

# Package ‘RStripe’

August 29, 2016

**Title** A Convenience Interface for the Stripe Payment API

**Version** 0.1

**Author** Evan Farrell <evandev@gastrograph.com>

**Maintainer** Evan Farrell <evandev@gastrograph.com>

**Description** A convenience interface for communicating with the Stripe payment processor to accept payments online. See <<https://stripe.com>> for more information.

**Depends** R (>= 3.1.1)

**Imports** jsonlite, RCurl, httr

**License** MIT + file LICENSE

**LazyData** true

**RoxygenNote** 5.0.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2016-07-15 00:28:26

## R topics documented:

rstripe . . . . .	3
stripe_balance_history . . . . .	3
stripe_cancel_transfer . . . . .	4
stripe_capture_charge . . . . .	4
stripe_close_dispute . . . . .	5
stripe_create_account_token . . . . .	5
stripe_create_application_fee_refund . . . . .	6
stripe_create_card . . . . .	7
stripe_create_card_token . . . . .	7
stripe_create_charge . . . . .	8
stripe_create_coupon . . . . .	9
stripe_create_customer . . . . .	10
stripe_create_invoice . . . . .	11
stripe_create_invoice_item . . . . .	11
stripe_create_plan . . . . .	12

stripe_create_recipient . . . . .	13
stripe_create_refund . . . . .	14
stripe_create_subscription . . . . .	14
stripe_create_transfer . . . . .	15
stripe_delete_card . . . . .	16
stripe_delete_coupon . . . . .	16
stripe_delete_customer . . . . .	17
stripe_delete_customer_discount . . . . .	17
stripe_delete_invoice_item . . . . .	18
stripe_delete_plan . . . . .	18
stripe_delete_recipient . . . . .	19
stripe_delete_subscription . . . . .	19
stripe_delete_subscription_discount . . . . .	20
stripe_list_application_fees . . . . .	20
stripe_list_application_fee_refunds . . . . .	21
stripe_list_cards . . . . .	21
stripe_list_charges . . . . .	22
stripe_list_coupons . . . . .	23
stripe_list_customers . . . . .	23
stripe_list_events . . . . .	24
stripe_list_invoices . . . . .	25
stripe_list_invoice_items . . . . .	25
stripe_list_plans . . . . .	26
stripe_list_recipients . . . . .	26
stripe_list_refunds . . . . .	27
stripe_list_subscriptions . . . . .	27
stripe_list_transfers . . . . .	28
stripe_pay_invoice . . . . .	28
stripe_retrieve_account . . . . .	29
stripe_retrieve_application_fee . . . . .	29
stripe_retrieve_application_fee_refund . . . . .	30
stripe_retrieve_balance . . . . .	30
stripe_retrieve_balance_transaction . . . . .	31
stripe_retrieve_card . . . . .	31
stripe_retrieve_charge . . . . .	32
stripe_retrieve_coupon . . . . .	32
stripe_retrieve_customer . . . . .	33
stripe_retrieve_event . . . . .	33
stripe_retrieve_invoice . . . . .	34
stripe_retrieve_invoice_item . . . . .	34
stripe_retrieve_invoice_line_items . . . . .	35
stripe_retrieve_plan . . . . .	35
stripe_retrieve_recipient . . . . .	36
stripe_retrieve_refund . . . . .	36
stripe_retrieve_subscription . . . . .	37
stripe_retrieve_transfer . . . . .	37
stripe_upcoming_customer_invoice . . . . .	38
stripe_update_application_fee_refund . . . . .	38

stripe_update_card . . . . .	39
stripe_update_charge . . . . .	40
stripe_update_coupon . . . . .	40
stripe_update_customer . . . . .	41
stripe_update_dispute . . . . .	42
stripe_update_invoice . . . . .	42
stripe_update_invoice_item . . . . .	43
stripe_update_plan . . . . .	44
stripe_update_recipient . . . . .	44
stripe_update_refund . . . . .	45
stripe_update_subscription . . . . .	46
stripe_update_transfer . . . . .	46

**Index** **48**

rstripe *rstripe.*

**Description**

rstripe.

stripe\_balance\_history  
*Retrieve a Stripe Account Balance History.*

**Description**

Retrieve the history of your account balance.

**Usage**

stripe\_balance\_history(api\_key, args = NULL)

**Arguments**

api_key	Your Stripe API Key
args	optional list which can include <b>available_on</b> <i>optional</i> : Filter on date with unix timestamp. <b>ending_before</b> <i>optional</i> : An object id which will show objects before <b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10 <b>source</b> <i>optional</i> : Only returns transactions associated with ID. <b>transfer</b> <i>optional</i> : Only returns transactions associated with transfer. <b>starting_after</b> <i>optional</i> : An object id which will show objects starting here <b>type</b> <i>optional</i> : Type: 'charge', 'refund', 'adjustment', 'application_fee', 'application_fee_refund', 'transfer', or 'transfer_failure' <b>currency</b> <i>optional</i> : 3 letter ISO

**Value**

A data frame with balance history information if succeeded.

---

```
stripe_cancel_transfer
```

*Cancel a transfer on Stripe.*

---

**Description**

This will cancel a transfer that is scheduled.

**Usage**

```
stripe_cancel_transfer(api_key, transfer_id)
```

**Arguments**

api_key	Your Stripe API Key
transfer_id	The id for the tranfer you want to cancel

**Value**

A data frame with nothing if successful.

---

```
stripe_capture_charge
```

*Capture a charge.*

---

**Description**

As part of the two step charge workflow, capture a charge. Read more about the workflow on the stripe website.

**Usage**

```
stripe_capture_charge(api_key, charge_id, args = NULL)
```

**Arguments**

api_key	Your Stripe API Key
charge_id	The identifier of the charge to be captured
args	an optional list that can include <b>amount</b> , <b>application_fee</b> <i>optional</i> A fee to add on to a charge. Only Stripe Connect <b>receipt_email</b> <i>optional</i> The email to send receipt to.

**Value**

Returns a charge data frame if the capture succeeded.

---

stripe\_close\_dispute *Close a dispute.*

---

**Description**

Close a dispute which you don't have evidence to.

**Usage**

```
stripe_close_dispute(api_key, charge_id)
```

**Arguments**

api_key	Your Stripe API Key
charge_id	The identifier of the charge with the dispute.

**Value**

Returns a dispute data frame if closed.

---

stripe\_create\_account\_token  
*Create a Bank Account Token*

---

**Description**

Create a bank account token to schedule a transfer to a recipient.

**Usage**

```
stripe_create_account_token(api_key, args)
```

**Arguments**

api_key	Your Stripe API Key
args	A list which must contain bank account information including <b>country required</b> Country bank account is in. Only 'US' supported. <b>routing_number required</b> routing number in string form. Not wire number. <b>account_number required</b> Account number for account. Must be checking.

**Value**

A data frame with the new bank account token info if successful.

**Examples**

```
## Not run: stripe_create_account_token("sk_test_BQokikJOvBiI2HlWgH4oIfQ2",
list(card=list(country="US", routing_number="110000000", account_number="0000000014343")))
## End(Not run)
```

---

```
stripe_create_application_fee_refund
```

*Create a refund for an Application Fee.*

---

**Description**

Create a refund for an application fee.

**Usage**

```
stripe_create_application_fee_refund(api_key, application_fee_id, args = NULL)
```

**Arguments**

`api_key` Your Stripe API Key

`application_fee_id`  
The id for the Application Fee you want to refund

`args` An optional list which can contain an amount to refund

**amount** *optional* default is entire fee

**metadata** *optional* A list which can contain anything to further describe.

**Value**

A data frame with refund information

**Examples**

```
## Not run: stripe_create_application_fee_refund("sk_test_BQokikJOvBiI2HlWgH4oIfQ2",
"fee_4cbC9iLv8PdUnk")
## End(Not run)
```

---

stripe\_create\_card     *Add a credit card to Stripe.*

---

### Description

Add a new credit card to a customer. Set it as default.

### Usage

```
stripe_create_card(api_key, customer_id, args)
```

### Arguments

api_key	Your Stripe API Key
customer_id	The id for the customer you want the credit card added to.
args	A required list of either a token, or a card dictionary. <a href="https://stripe.com/docs/api#create_card">/hrefhttps://stripe.com/docs/api#create_card</a> more info here

### Value

A data frame with card information

### Examples

```
## Not run: stripe_create_card("sk_test_BQokikJOvBiI2H1WgH4o1fQ2", "cus_4cb7HNjNSqqV2H",  
list(card="tok_14SsjM2eZvKYlo2CCtdx2Av0")  
## End(Not run)  
  
## Not run: stripe_create_card("sk_test_BQokikJOvBiI2H1WgH4o1fQ2", "cus_4cb7HNjNSqqV2H",  
list(card=list(number="4242424242424242", exp_month="03", exp_year="2018")))  
## End(Not run)
```

---

stripe\_create\_card\_token  
                          *Create a Card Token*

---

### Description

Create a card token to be used by other functions. You can use this to create a charge, or to attach to a customer.

### Usage

```
stripe_create_card_token(api_key, args)
```

**Arguments**

api_key	Your Stripe API Key
args	A list which must contain card information including <b>number required</b> <b>cvc required</b> <b>exp_month required</b> <b>exp_year required</b> <b>address_city optional</b> <b>address_country optional</b> <b>address_line1 optional</b> <b>address_line2 optional</b> <b>address_state optional</b> <b>address_zip optional</b> <b>name optional</b> <b>customer optional</b> Customer to attach token to.

**Value**

A data frame with the new card token info if successful.

**Examples**

```
## Not run: stripe_create_card_token("sk_test_BQokikJOvBiI2H1WgH4oIfQ2",
list(card=list(number="4242424242424242", exp_month="03", exp_year="2018")))
## End(Not run)
```

---

stripe\_create\_charge *Create a charge.*

---

**Description**

Create a new charge for a customer or standalone on card.

**Usage**

```
stripe_create_charge(api_key, args)
```

**Arguments**

api_key	Your Stripe API Key
args	A list of arguments <b>amount required</b> Amount for transfer. <b>currency required</b> The 3-letter currency.

**customer** *optional* Either customer or card required.  
**card** *optional* Either customer or card required.  
**description** *optional* The description of the transfer.  
**capture** *optional* True/false to capture charge later.  
**recipient\_email** *optional* The email to send receipt to.  
**metadata** *optional* A list which can contain anything to further describe.  
**statement\_description** *optional* 15 character description to appear on statement.  
**application\_fee** *optional* A fee in cents applied to charge.

### Value

A data frame with the new charge info if successful.

### Examples

```
## Not run: stripe_create_charge("sk_test_BQokikJOvBiI2H1WgH4oIfQ2",
list(card="tok_14SsjM2eZvKY1o2CCtdx2Av0", amount=3000, currency="usd",
metadata=list(order_id="12545")))
## End(Not run)
```

---

stripe\_create\_coupon *Create a coupon.*

---

### Description

Create a new coupon in your account.

### Usage

```
stripe_create_coupon(api_key, args)
```

### Arguments

api_key	Your Stripe API Key
args	An list
<b>duration</b>	<b>required</b> Can be <i>forever</i> , <i>once</i> , <i>repeating</i>
<b>id</b>	<b>optional</b> An id to identify. e.g. <i>FALL25OFF</i>
<b>amount_off</b>	<b>optional</b> required if <i>percent_off</i> is not passed. Integer.
<b>currency</b>	<b>optional</b> required if amount_off is passed.
<b>duration_in_months</b>	<b>optional</b> required if duration is repeating.
<b>max_redemptions</b>	<b>optional</b> total amount of times to be used.
<b>metadata</b>	<i>optional</i> A list which can contain anything to further describe.
<b>percent_off</b>	<b>optional</b> required if amount_off not passed.
<b>redeem_by</b>	<b>optional</b> unix timestamp that it ends.

**Value**

A data frame with the coupon information

**Examples**

```
## Not run: stripe_create_coupon("sk_test_BQokikJ0vBiI2HlWgH4o1fQ2",
list(amount_off=200, duration='forever'))
## End(Not run)
```

---

stripe\_create\_customer

*Create a Customer.*

---

**Description**

Create a new customer and add information to them.

**Usage**

```
stripe_create_customer(api_key, args = NULL)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>args</code>	an optional list which can contain <ul style="list-style-type: none"> <li><b>account_balance</b> <i>optional</i> In cents starting account balance. Positive means amount to be paid off. Negative is credit used towards charge.</li> <li><b>card</b> <i>optional</i> Either token from Stripe.js or card details.</li> <li><b>description</b> <i>optional</i> The description of the transfer.</li> <li><b>metadata</b> <i>optional</i> A list which can contain anything to further describe.</li> <li><b>coupon</b> <i>optional</i> A coupon code for a discount applied on all charges.</li> <li><b>email</b> <i>optional</i> Customer's email address.</li> <li><b>plan</b> <i>optional</i> A plan id to subscribe the customer to.</li> <li><b>quantity</b> <i>optional</i> The amount of plans to subscribe them to.</li> <li><b>trial_end</b> <i>optional</i> Timestamp for end of trial period. Overrides plan trial.</li> </ul>

**Value**

A data frame with the new Customer info if successful.

---

stripe\_create\_invoice *Create an Invoice.*

---

### Description

Create a new invoice for a customer.

### Usage

```
stripe_create_invoice(api_key, args)
```

### Arguments

api_key	Your Stripe API Key
args	A list which must contain <b>customer</b> <b>required</b> customer id for the invoice. <b>description</b> <i>optional</i> description of invoice. <b>application_fee</b> <i>optional</i> fee in cents to apply to invoice. <b>metadata</b> <i>optional</i> A list which can contain anything to further describe. <b>subscription</b> <i>optional</i> id of the subscription to invoice. <b>statement_description</b> <i>optional</i> Goes on credit card statement of customer.

### Value

A data frame with the new invoice if successful.

---

stripe\_create\_invoice\_item  
*Create an Invoice Item.*

---

### Description

Create a new invoice item with amount and attach it to a subscription or invoice. If you leave invoice blank it will be added to the next upcoming invoice.

### Usage

```
stripe_create_invoice_item(api_key, args)
```

**Arguments**

api_key	Your Stripe API Key
args	A list which must contain <ul style="list-style-type: none"> <li><b>amount required</b> In cents how much for item.</li> <li><b>currency required</b> 3-letter ISO code for currency.</li> <li><b>description</b> <i>optional</i> A string to attach to item for easy tracking.</li> <li><b>metadata</b> <i>optional</i> key/value pairs that you can attach to an invoice item.</li> <li><b>invoice</b> <i>optional</i> Add to this invoice or leave blank for next invoice.</li> <li><b>subscription</b> <i>optional</i> Add to a specific subscriptions invoice.</li> </ul>

**Value**

A data frame with the new invoice item if successful.

---

stripe_create_plan	<i>Create a Plan</i>
--------------------	----------------------

---

**Description**

Create a new plan on your stripe account.

**Usage**

```
stripe_create_plan(api_key, args)
```

**Arguments**

api_key	Your Stripe API Key
args	A list which must contain <ul style="list-style-type: none"> <li><b>id</b> The id of the plan to use. e.g. 'gold'</li> <li><b>amount required</b> The amount in cents to charge.</li> <li><b>currency required</b> 3-letter ISO code for currency.</li> <li><b>interval required</b> Specifies billing frequency. <b>day, week month, year.</b></li> <li><b>name required</b> The name of plan. Appears on invoice.</li> <li><b>metadata</b> <i>optional</i> A list which can contain anything to further describe.</li> <li><b>trial_period_days</b> <i>optional</i> Specifies trial period in day.</li> <li><b>interval_count</b> <i>optional</i> intervals in billing. e.g. 3 if interval=month</li> <li><b>statement_description</b> <i>optional</i> String to appear on invoice. Up to 15 characters</li> </ul>

**Value**

A data frame with the new plan info if successful.

**Examples**

```
## Not run: stripe_create_plan("sk_test_BQokikJOvBiI2HlWgH4o1fQ2",
list(id="example-gold", amount=3000, currency="usd", interval="month"))
## End(Not run)
```

---

```
stripe_create_recipient
```

*Add a recipient to Stripe.*

---

**Description**

Add a new recipient to your account to schedule transfers and more.

**Usage**

```
stripe_create_recipient(api_key, args)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>args</code>	A required list that must contain <ul style="list-style-type: none"> <li><b>name required</b> The legal name of recipient.</li> <li><b>type required</b> put 'individual' or 'corporation'.</li> <li><b>tax_id optional</b> The tax ID either 'SSN' for individual or 'EIN' for corporation</li> <li><b>bank_account optional</b> token or dictionary of details.</li> <li><b>card optional</b> A US Visa/Mastercard debit card to put transfer on.</li> <li><b>email optional</b> Recipients email address.</li> <li><b>description optional</b> Arbitrary string to describe.</li> <li><b>metadata optional</b> key/value for storing information.</li> </ul>

**Value**

A data frame with new recipient information

---

stripe\_create\_refund *Create a Refund*

---

### Description

Create a new refund on a charge.

### Usage

```
stripe_create_refund(api_key, charge_id, args = NULL)
```

### Arguments

api_key	Your Stripe API Key
charge_id	The id of the charge to refund.
args	A list can contain <b>amount</b> , <b>amount</b> <i>required</i> An amount in cents to refund up to remaining charge. <b>refund_application_fee</b> <i>required</i> true/false if fee should be refunded. If true, proportional amount of refund is refunded. <b>metadata</b> <i>required</i> Any metadata to include on the refund.

### Value

A data frame with the new refund info if successful.

---

stripe\_create\_subscription  
*Create a Subscription*

---

### Description

Create a new subscription for a customer and add it to them.

### Usage

```
stripe_create_subscription(api_key, customer_id, args)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>customer_id</code>	The customer id you want to add a subscription to.
<code>args</code>	A list which includes: <ul style="list-style-type: none"> <li><b>plan required</b> The plan id</li> <li><b>coupon optional</b> A coupon to apply to this subscription.</li> <li><b>trial_end optional</b> When the trial ends. Unix timestamp to override plan. You can use <b>now</b> to make it end now.</li> <li><b>card optional</b> A token or card dictionary to use instead of default. <b>required</b> if customer has no card.</li> <li><b>quantity optional</b> Quantity to place in subscription.</li> <li><b>application_fee_percent optional</b> A decimal 1 to 100 for percentage of amount.</li> <li><b>metadata optional</b> A list which can contain anything to further describe.</li> </ul>

**Value**

A data frame with the new subscription info if successful.

---

```
stripe_create_transfer
```

*Create a transfer to Stripe.*

---

**Description**

Create a new transfer that will be first marked as pending.

**Usage**

```
stripe_create_transfer(api_key, args)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>args</code>	A list of arguments <ul style="list-style-type: none"> <li><b>amount required</b> Amount for transfer.</li> <li><b>currency required</b> The 3-letter currency.</li> <li><b>recipient required</b> The recipient id of the transfer.</li> <li><b>description optional</b> The description of the transfer.</li> <li><b>metadata optional</b> A list which can contain anything to further describe.</li> <li><b>bank_account optional</b> The bank to transfer money to. Include this, card, or neither.</li> <li><b>card optional</b> The card to transfer the money to. Include either Bank Account Card, or neither.</li> <li><b>statement_description optional</b> 15 character description to appear on statement.</li> </ul>

**Value**

A data frame with transfer information

---

stripe\_delete\_card     *Delete a credit card on Stripe.*

---

**Description**

Delete a credit card off a customer's account.

**Usage**

```
stripe_delete_card(api_key, customer_id, card_id)
```

**Arguments**

api_key	Your Stripe API Key
customer_id	The id for the customer where the card is.
card_id	The id for the card you want to delete

**Value**

A data frame with card information

---

stripe\_delete\_coupon     *Delete a coupon.*

---

**Description**

Delete a specific coupon from your account.

**Usage**

```
stripe_delete_coupon(api_key, coupon_id)
```

**Arguments**

api_key	Your Stripe API Key
coupon_id	The coupon you want to delete

**Value**

A null dataframe if succeeded.

---

`stripe_delete_customer`*Delete a Customer.*

---

**Description**

Remove a customer from your stripe account.

**Usage**

```
stripe_delete_customer(api_key, customer_id)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>customer_id</code>	The id of the customer you want to delete

**Value**

A data frame empty if successful.

---

`stripe_delete_customer_discount`*Delete a customer discount.*

---

**Description**

Delete a discount from a customer.

**Usage**

```
stripe_delete_customer_discount(api_key, customer_id)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>customer_id</code>	The identifier of the customer who has the discount to be deleted.

**Value**

Returns a data frame deleted true if if succeeded.

stripe\_delete\_invoice\_item

*Delete an Invoice Item.*

---

**Description**

Remove an invoice item from an invoice.

**Usage**

```
stripe_delete_invoice_item(api_key, item_id)
```

**Arguments**

api_key	Your Stripe API Key
item_id	The invoice item id you want to delete

**Value**

A data frame without errors if successful.

---

stripe\_delete\_plan *Delete a plan on Stripe.*

---

**Description**

Delete a plan in your system. Only able to do this if you have no subscriptions on it.

**Usage**

```
stripe_delete_plan(api_key, plan_id)
```

**Arguments**

api_key	Your Stripe API Key
plan_id	The id for the plan you want to delete

**Value**

A data frame with nothing if successful.

---

```
stripe_delete_recipient
```

*Delete a recipient on Stripe.*

---

**Description**

Remove a recipient from your stripe account.

**Usage**

```
stripe_delete_recipient(api_key, recipient_id)
```

**Arguments**

api_key	Your Stripe API Key
recipient_id	The id for the recipient you want to delete

**Value**

A data frame with recipient information

---

```
stripe_delete_subscription
```

*Delete a subscription on Stripe.*

---

**Description**

Delete a subscription on a particular customer.

**Usage**

```
stripe_delete_subscription(api_key, customer_id, subscription_id)
```

**Arguments**

api_key	Your Stripe API Key
customer_id	The id of the customer with the subscription.
subscription_id	The id for the subscription you want to delete

**Value**

A data frame with nothing if successful.

---

```
stripe_delete_subscription_discount
    Delete a subscription discount.
```

---

**Description**

Delete a discount from a customer's subscription.

**Usage**

```
stripe_delete_subscription_discount(api_key, customer_id, subscription_id)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>customer_id</code>	The identifier of the customer who has the discount to be deleted.
<code>subscription_id</code>	The identifier of the subscription.

**Value**

Returns a data frame deleted true if if succeeded.

---

```
stripe_list_application_fees
    List all Application Fees.
```

---

**Description**

List all the application fees. Can filter by charge, date, etc.

**Usage**

```
stripe_list_application_fees(api_key, args = NULL)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>args</code>	optional list that can include: <ul style="list-style-type: none"> <li><b>charge</b> <i>optional</i> Only return feew associated with this charge.</li> <li><b>ending_before</b> <i>optional</i>: An object id which will show objects before</li> <li><b>limit</b> <i>optional</i>: A number 1 to 100 to limit the items. Default is 10</li> <li><b>starting_after</b> <i>optional</i>: An object id which will show objects starting here</li> </ul>

**Value**

A data frame with the application fees you have created that allow you to refund them.

---

`stripe_list_application_fee_refunds`*List all refunds for an Application Fee.*

---

**Description**

List all refunds associated with a specific application fee.

**Usage**

```
stripe_list_application_fee_refunds(api_key, application_fee_id, args = NULL)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>application_fee_id</code>	The id for the Application Fee you want to refund
<code>args</code>	An optional list which can contain <b>ending_before</b> <i>optional</i> : An object id which will show objects before <b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10 <b>starting_after</b> <i>optional</i> : An object id which will show objects starting here

**Value**

A data frame with application fee refund object if succeeded

**Examples**

```
## Not run: stripe_list_application_fee_refunds("sk_test_BQokikJOvBiI2HlWgH4oIfQ2", "fee_4cbC9iLv8PdUnk",  
list(limit=3))  
## End(Not run) #returns 3 refunds
```

---

`stripe_list_cards`      *List credit card on customers.*

---

**Description**

List all the credit cards linked to a customer.

**Usage**

```
stripe_list_cards(api_key, customer_id, args = NULL)
```

**Arguments**

api_key	Your Stripe API Key
customer_id	The id for the customer you want to list the cards
args	an optional list that can contain <b>ending_before</b> <i>optional</i> : An object id which will show objects before <b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10 <b>starting_after</b> <i>optional</i> : An object id which will show objects starting here

**Value**

A data frame with card information

---

stripe\_list\_charges    *List all charges.*

---

**Description**

List all charges conducted in a time frame or on customer. Limits to 10 originally.

**Usage**

```
stripe_list_charges(api_key, args = NULL)
```

**Arguments**

api_key	Your Stripe API Key
args	an optional list that can include <b>customer</b> , <b>customer</b> <i>optional</i> A customer id to show only their charges. <b>ending_before</b> <i>optional</i> An id to show the ones before this item. <b>limit</b> <i>optional</i> The limit to show. Can be 1 to 100. Default is 10. <b>starting_after</b> <i>optional</i> An id to show the ones starting after this item.

**Value**

Returns a data frame with all customer charges. 10 if no limit specified.

---

stripe\_list\_coupons *List all coupons.*

---

### Description

List all the coupons currently in your account.

### Usage

```
stripe_list_coupons(api_key, args = NULL)
```

### Arguments

**api\_key** Your Stripe API Key

**args** An optional list which can contain

- ending\_before** *optional*: An object id which will show objects before
- limit** *optional*: A number 1 to 100 to limit the items. Default is 10
- starting\_after** *optional*: An object id which will show objects starting here

### Value

A data frame with all the coupons.

### Examples

```
## Not run: stripe_list_application_fee_refunds("sk_test_BQokikJOvBiI2HlWgH4o1fQ2", "fee_4cbC9iLv8PdUnk",  
list(limit=3))  
## End(Not run) #returns 3 refunds
```

---

stripe\_list\_customers *Retrieve a list of Customer.*

---

### Description

Retrieve a list of all customers. Defaults up to 10.

### Usage

```
stripe_list_customers(api_key, args = NULL)
```

**Arguments**

api_key	Your Stripe API Key
args	an optional list of arguments. Can include <b>ending_before</b> <i>optional</i> : An object id which will show objects before <b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10 <b>starting_after</b> <i>optional</i> : An object id which will show objects starting here

**Value**

A data frame with the Customer list if successful.

---

stripe\_list\_events      *List all events.*

---

**Description**

list all the events that have happened.

**Usage**

```
stripe_list_events(api_key, args = NULL)
```

**Arguments**

api_key	Your Stripe API Key
args	an optional list that can include: <b>ending_before</b> <i>optional</i> : An object id which will show objects before <b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10 <b>starting_after</b> <i>optional</i> : An object id which will show objects starting here <b>type</b> <i>optional</i> Can contain specific event name or group with '*' wildcard.

**Value**

Returns an event data frame containing all the events if valid.

---

stripe\_list\_invoices *List Invoices.*

---

### Description

List invoices for a particular customer or all customers within a date.

### Usage

```
stripe_list_invoices(api_key, args = NULL)
```

### Arguments

api_key	Your Stripe API Key
args	An optional list that can include: <b>ending_before</b> <i>optional</i> : An object id which will show objects before <b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10 <b>starting_after</b> <i>optional</i> : An object id which will show objects starting here

### Value

A data frame with the invoices if successful.

---

stripe\_list\_invoice\_items  
*List all Invoice Item.*

---

### Description

List all the invoice items in your account. This can be limited by customer, date, etc.

### Usage

```
stripe_list_invoice_items(api_key, args = NULL)
```

### Arguments

api_key	Your Stripe API Key
args	A list which can contain: <b>ending_before</b> <i>optional</i> : An object id which will show objects before <b>customer</b> <i>optional</i> : The customer ID whose invoice items to return <b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10 <b>starting_after</b> <i>optional</i> : An object id which will show objects starting here

### Value

A data frame with the the invoice items if successful.

---

```
stripe_list_plans      List Plans.
```

---

**Description**

List all the plans in your system. It defaults to showing 10 unless you change the limit.

**Usage**

```
stripe_list_plans(api_key, args = NULL)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>args</code>	an optional list that can contain
	<b>ending_before</b> <i>optional</i> : An object id which will show objects before
	<b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10
	<b>starting_after</b> <i>optional</i> : An object id which will show objects starting here

**Value**

A data frame with plan information

---

```
stripe_list_recipients
      List all current recipients.
```

---

**Description**

List all the current recipients in your account.

**Usage**

```
stripe_list_recipients(api_key, args = NULL)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>args</code>	an optional list that can contain:
	<b>verified</b> <i>optional</i> : Show only verified recipients.
	<b>ending_before</b> <i>optional</i> : An object id which will show objects before
	<b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10
	<b>starting_after</b> <i>optional</i> : An object id which will show objects starting here

**Value**

A data frame with recipient's information

---

```
stripe_list_refunds List Refunds
```

---

### Description

List all the refunds that a charge has. You can have multiple refunds per charge until the full amount is reached.

### Usage

```
stripe_list_refunds(api_key, charge_id, args = NULL)
```

### Arguments

<code>api_key</code>	Your Stripe API Key
<code>charge_id</code>	The id of the charge with the refund.
<code>args</code>	an optional list that can contain: <ul style="list-style-type: none"> <li><b>ending_before</b> <i>optional</i>: An object id which will show objects before</li> <li><b>limit</b> <i>optional</i>: A number 1 to 100 to limit the items. Default is 10</li> <li><b>starting_after</b> <i>optional</i>: An object id which will show objects starting here</li> </ul>

### Value

A data frame with refunds information

---

```
stripe_list_subscriptions
List Subscriptions
```

---

### Description

List all the subscriptions currently acted on a customer.

### Usage

```
stripe_list_subscriptions(api_key, customer_id, args = NULL)
```

### Arguments

<code>api_key</code>	Your Stripe API Key
<code>customer_id</code>	The id of the customer with the subscriptions.
<code>args</code>	an optional list that can contain <ul style="list-style-type: none"> <li><b>ending_before</b> <i>optional</i>: An object id which will show objects before</li> <li><b>limit</b> <i>optional</i>: A number 1 to 100 to limit the items. Default is 10</li> <li><b>starting_after</b> <i>optional</i>: An object id which will show objects starting here</li> </ul>

**Value**

A data frame with subscriptions information

---

stripe\_list\_transfers *List Transfers.*

---

**Description**

List all the Bank Account transfers for the account.

**Usage**

```
stripe_list_transfers(api_key, args = NULL)
```

**Arguments**

api_key	Your Stripe API Key
args	an optional list that can contain: <b>ending_before</b> <i>optional</i> : An object id which will show objects before <b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10 <b>recipient</b> <i>optional</i> : The id of a recipient to limit to. <b>starting_after</b> <i>optional</i> : An object id which will show objects starting here <b>status</b> <i>optional</i> : transfers with: <b>pending, paid, failed</b>

**Value**

A data frame with transfers information

---

stripe\_pay\_invoice *Pay an Invoice.*

---

**Description**

Pay the invoice right now with customer information.

**Usage**

```
stripe_pay_invoice(api_key, invoice_id)
```

**Arguments**

api_key	Your Stripe API Key
invoice_id	The invoice id you want to pay

**Value**

A data frame with the invoice if successful.

---

```
stripe_retrieve_account
```

*Your Stripe Account.*

---

**Description**

Retrieve your account.

**Usage**

```
stripe_retrieve_account(api_key)
```

**Arguments**

api\_key            Your Stripe API key

**Value**

Your account information in a data frame.

**Examples**

```
## Not run: stripe_retrieve_account("sk_test_2337VDxzNLS332zss0")
```

---

```
stripe_retrieve_application_fee
```

*Retrieve an Application Fee.*

---

**Description**

Retrieve the data behind a specific applicaiton fee.

**Usage**

```
stripe_retrieve_application_fee(api_key, application_fee_id)
```

**Arguments**

api\_key            Your Stripe API Key  
application\_fee\_id    The id for the Application Fee you want to retrieve

**Value**

A data frame with refund app fee information

stripe\_retrieve\_application\_fee\_refund

*Retrieve an existing refund for an Application Fee.*

---

**Description**

Retrieve the information of a refund.

**Usage**

```
stripe_retrieve_application_fee_refund(api_key, application_fee_id, refund_id)
```

**Arguments**

api_key	Your Stripe API Key
application_fee_id	The id for the Application Fee you want to refund
refund_id	The id for the Refund you want to retrieve

**Value**

A data frame with refund information

---

stripe\_retrieve\_balance

*Retrieve Stripe Account Balance.*

---

**Description**

Retrieve the balance of your stripe account.

**Usage**

```
stripe_retrieve_balance(api_key)
```

**Arguments**

api_key	Your Stripe API Key
---------	---------------------

**Value**

A data frame with your account balance

---

```
stripe_retrieve_balance_transaction
```

*Retrieve a Stripe Account Balance transaction.*

---

**Description**

Retrieve information about a transaction on your account.

**Usage**

```
stripe_retrieve_balance_transaction(api_key, transaction_id)
```

**Arguments**

`api_key` Your Stripe API Key  
`transaction_id` A stripe you want to get information about.

**Value**

A data frame with balance transaction information if succeeded.

---

```
stripe_retrieve_card
```

*Retrieve a card from a customer*

---

**Description**

Retrieve the information about a credit card. Can retrieve last 4 digits.

**Usage**

```
stripe_retrieve_card(api_key, card_id, customer_id)
```

**Arguments**

`api_key` Your Stripe API Key  
`card_id` The id for the card you want to retrieve  
`customer_id` The id for the customer where the card is added to.

**Value**

A data frame with card information

stripe\_retrieve\_charge

*Retrieve a charge.*

---

### Description

Retrieve all the information about a specific charge.

### Usage

```
stripe_retrieve_charge(api_key, charge_id)
```

### Arguments

api_key	Your Stripe API Key
charge_id	The identifier of the charge to be retrieved

### Value

Returns a charge data frame if valid.

### Examples

```
## Not run: stripe_retrieve_charge("sk_test_BQokikJOvBiI2HlWgH4o1fQ2",  
  "ch_14TLyd2eZvKY1o2CdryxwrF8")  
## End(Not run)
```

---

stripe\_retrieve\_coupon

*Retrieve a coupon.*

---

### Description

Retrieve the information about a coupon.

### Usage

```
stripe_retrieve_coupon(api_key, coupon_id)
```

### Arguments

api_key	Your Stripe API Key
coupon_id	The coupon you want to retrieve.

### Value

A data frame with the coupon information

---

`stripe_retrieve_customer`*Retrieve a Customer.*

---

**Description**

Retrieve the information about a specific customer using their id.

**Usage**

```
stripe_retrieve_customer(api_key, customer_id)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>customer_id</code>	The id of the customer you want to retrieve

**Value**

A data frame with the Customer info if successful.

---

`stripe_retrieve_event` *Retrieve an event.*

---

**Description**

Retrieve the information about a specific event.

**Usage**

```
stripe_retrieve_event(api_key, event_id)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>event_id</code>	The identifier of the event to be retrieved

**Value**

Returns an event data frame if valid.

---

`stripe_retrieve_invoice`*Retrieve an Invoice.*

---

**Description**

Retrieve all the information on a particular invoice.

**Usage**

```
stripe_retrieve_invoice(api_key, invoice_id)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>invoice_id</code>	The invoice id you want to retrieve

**Value**

A data frame with the invoice if successful.

---

`stripe_retrieve_invoice_item`*Retrieve an Invoice Item.*

---

**Description**

Retrieve a specific invoice item's info using it's id.

**Usage**

```
stripe_retrieve_invoice_item(api_key, item_id)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>item_id</code>	The invoice item id you want to retrieve

**Value**

A data frame with the invoice item if successful.

---

stripe\_retrieve\_invoice\_line\_items  
*Retrieve Invoice Items.*

---

**Description**

Retrieve all the items on an invoice.

**Usage**

```
stripe_retrieve_invoice_line_items(api_key, invoice_id, args = NULL)
```

**Arguments**

api_key	Your Stripe API Key
invoice_id	The invoice id with the items you want to retrieve
args	An optional list that can include <b>ending_before</b> <i>optional</i> : An object id which will show objects before <b>limit</b> <i>optional</i> : A number 1 to 100 to limit the items. Default is 10 <b>starting_after</b> <i>optional</i> : An object id which will show objects starting here

**Value**

A data frame with the invoice items if successful.

---

stripe\_retrieve\_plan *Retrieve a Plan*

---

**Description**

Retrieve the information on a particular plan on your account.

**Usage**

```
stripe_retrieve_plan(api_key, plan_id)
```

**Arguments**

api_key	Your Stripe API Key
plan_id	The id of the plan you want to retrieve

**Value**

A data frame with the plan info if successful.

---

`stripe_retrieve_recipient`*Retrieve information on a recipient.*

---

**Description**

Retrieve all the information on a recipient.

**Usage**

```
stripe_retrieve_recipient(api_key, recipient_id)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>recipient_id</code>	The id for the recipient you want to retrieve.

**Value**

A data frame with recipient information

---

`stripe_retrieve_refund`*Retrieve a refund.*

---

**Description**

Retrieve all the information about a particular refund on a charge.

**Usage**

```
stripe_retrieve_refund(api_key, charge_id, refund_id)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>charge_id</code>	The id of the charge with the refund.
<code>refund_id</code>	The id of the refund you want to retrieve

**Value**

A data frame with the refund info if successful.

---

stripe\_retrieve\_subscription  
*Retrieve a Subscription*

---

**Description**

Retrieve all the information on a subscription for a customer.

**Usage**

```
stripe_retrieve_subscription(api_key, customer_id, subscription_id)
```

**Arguments**

api_key	Your Stripe API Key
customer_id	The id of the customer with the subscription.
subscription_id	The id of the subscription you want to retrieve

**Value**

A data frame with the subscription info if successful.

---

stripe\_retrieve\_transfer  
*Retrieve a transfer*

---

**Description**

You can get all the information about a pending transfer or an accepted transfer.

**Usage**

```
stripe_retrieve_transfer(api_key, transfer_id)
```

**Arguments**

api_key	Your Stripe API Key
transfer_id	The id of the transfer you want to retrieve

**Value**

A data frame with the transfer info if successful.

---

stripe\_upcoming\_customer\_invoice

*Retrieve upcoming customer's invoice.*

---

### Description

Retrieve the upcoming customer invoice to send out.

### Usage

```
stripe_upcoming_customer_invoice(api_key, customer_id)
```

### Arguments

api_key	Your Stripe API Key
customer_id	The customer id that you want the upcoming invoice

### Value

A data frame with the upcoming invoice if successful.

---

stripe\_update\_application\_fee\_refund

*Update a refund for an Application Fee.*

---

### Description

Update all the information of a refund of an application fee.

### Usage

```
stripe_update_application_fee_refund(api_key, application_fee_id, refund_id,  
args)
```

### Arguments

api_key	Your Stripe API Key
application_fee_id	The id for the Application Fee you want to refund
refund_id	The id for the Refund you want to update
args	An optional list which can contain: <b>metadata</b> <i>optional</i> A list which can contain anything to further describe.

**Value**

A data frame with application fee refund object if succeeded

**Examples**

```
## Not run: stripe_create_application_fee_refund("sk_test_BQokikJOvBiI2HlWgH4o1fQ2", "fee_4cbC9iLv8PdUnk",
list(metadata=list(reason="Accidental charge")))
## End(Not run)
```

---

stripe\_update\_card      *Update a credit card on Stripe.*

---

**Description**

Update information about the billing address for a credit card as well as other fields.

**Usage**

```
stripe_update_card(api_key, customer_id, card_id, args)
```

**Arguments**

api_key	Your Stripe API Key
customer_id	The id for the customer you want the credit card added to.
card_id	The id for the card you want to update
args	A list that can contain the following
	<b>address_city</b> <i>optional</i>
	<b>address_country</b> <i>optional</i>
	<b>address_line1</b> <i>optional</i>
	<b>address_line2</b> <i>optional</i>
	<b>address_state</b> <i>optional</i>
	<b>address_zip</b> <i>optional</i>
	<b>exp_month</b> <i>optional</i>
	<b>exp_year</b> <i>optional</i>
	<b>name</b> <i>optional</i>

**Value**

A data frame with card information

**Examples**

```
## Not run: stripe_update_card("sk_test_BQokikJOvBiI2HlWgH4o1fQ2", "cus_4cb7HNjNSqqV2H",
"card_14TLyd2eZvKY1oc2C9PXnFmi", list(exp_year="04"))
## End(Not run)
```

---

stripe\_update\_charge *Update a charge.*

---

### Description

Update the information about a charge.

### Usage

```
stripe_update_charge(api_key, charge_id, args)
```

### Arguments

api_key	Your Stripe API Key
charge_id	The identifier of the charge to be updated
args	A list that can include: <b>metadata</b> <i>optional</i> A list of metadata to include <b>description</b> <i>optional</i> A string to describe it

### Value

Returns a charge data frame if succeeded.

### Examples

```
## Not run: stripe_update_charge("sk_test_BQokikJOvBiI2HlWgH4oIfQ2",  
"ch_14TLyd2eZvKYlo2CdryxwrF8", list(description="for the item",  
metadata=list(order_id="3453")))  
## End(Not run)
```

---

stripe\_update\_coupon *Update a coupon.*

---

### Description

Update the information about a coupon.

### Usage

```
stripe_update_coupon(api_key, coupon_id, args)
```

**Arguments**

api_key	Your Stripe API Key
coupon_id	The coupon you want to update.
args	An list which can contain: <b>description</b> <i>optional</i> The description of the transfer. <b>metadata</b> <i>optional</i> A list which can contain anything to further describe.

**Value**

A data frame with the new coupon information if succeeded.

---

stripe\_update\_customer

*Update a Customer.*

---

**Description**

Update the customer's information or add a new card to a customer.

**Usage**

```
stripe_update_customer(api_key, customer_id, args)
```

**Arguments**

api_key	Your Stripe API Key
customer_id	The id of the customer you want to update
args	an optional list which can contain: <b>account_balance</b> <i>optional</i> In cents starting account balance. Positive means amount to be paid off. Negative is credit used towards charge. <b>card</b> <i>optional</i> Either token from Stripe.js or card details. <b>description</b> <i>optional</i> The description of the transfer. <b>default_card</b> <i>optional</i> The id of a card to use as default. <b>metadata</b> <i>optional</i> A list which can contain anything to further describe. <b>coupon</b> <i>optional</i> A coupon code for a discount applied on all charges. <b>email</b> <i>optional</i> Customer's email address.

**Value**

A data frame with the updated Customer info if successful.

stripe\_update\_dispute *Update a dispute.*

---

### Description

Update a dispute and add evidence to it.

### Usage

```
stripe_update_dispute(api_key, charge_id, args)
```

### Arguments

api_key	Your Stripe API Key
charge_id	The identifier of the charge with the dispute.
args	An optional list that can contain <b>evidence</b> <i>optional</i> Evidence towards resolving dispute. <b>metadata</b> <i>optional</i> key/value for storing dispute information.

### Value

Returns a dispute data frame if valid.

### Examples

```
## Not run: stripe_update_dispute("sk_test_BQokikJOvBiI2HlWgH4o1fQ2",  
"ch_14TLyd2eZvKYlo2CdryxwrF8", list(evidence="Here's evidence showing  
this charge is legitimate.",metadata=list(exhibitA="This happened")))  
## End(Not run)
```

---

stripe\_update\_invoice *Update an Invoice.*

---

### Description

Update the information on an invoice, add a fee, close, or forgive it.

### Usage

```
stripe_update_invoice(api_key, invoice_id, args)
```

**Arguments**

api_key	Your Stripe API Key
invoice_id	The invoice id you want to update
args	Can include: <b>application_fee</b> <b>description</b> <b>closed</b> <b>forgiven</b> <b>statement_description</b> <b>metadata</b>

**Value**

A data frame with the updated invoice if successful.

---

```
stripe_update_invoice_item
      Update an Invoice Item.
```

---

**Description**

Update the information or amount of an invoice item.

**Usage**

```
stripe_update_invoice_item(api_key, item_id, args)
```

**Arguments**

api_key	Your Stripe API Key
item_id	The invoice item id you want to update
args	Can include <b>amount</b> <i>optional</i> Amount in cents for item. <b>description</b> <i>optional</i> A string to attach to item for easy tracking. <b>metadata</b> <i>optional</i> key/value pairs that you can attach to an invoice item.

**Value**

A data frame with the updated invoice item if successful.

---

stripe\_update\_plan      *Update a Plan*

---

### Description

Update the information about a particular plan on your stripe account.

### Usage

```
stripe_update_plan(api_key, plan_id, args)
```

### Arguments

api_key	Your Stripe API Key
plan_id	The id of the plan you want to update
args	A list can contain <ul style="list-style-type: none"> <li><b>name</b> <i>optional</i> The name of plan. Appears on invoice.</li> <li><b>metadata</b> <i>optional</i> A list which can contain anything to further describe.</li> <li><b>statement_description</b> <i>optional</i> String to appear on invoice. Up to 15 characters</li> </ul>

### Value

A data frame with the updated plan info if successful.

---

stripe\_update\_recipient  
*Update a recipient on Stripe.*

---

### Description

Update or add information about a recipient on stripe.

### Usage

```
stripe_update_recipient(api_key, recipient_id, args)
```

**Arguments**

api_key	Your Stripe API Key
recipient_id	The id for the recipient you want to update.
args	A list that can contain <ul style="list-style-type: none"> <li><b>name</b> <i>optional</i> The legal name of recipient.</li> <li><b>tax_id</b> <i>optional</i> The tax ID either 'SSN' for individual or 'EIN' for corporation</li> <li><b>bank_account</b> <i>optional</i> token or dictionary of details.</li> <li><b>card</b> <i>optional</i> A US Visa/Mastercard debit card to put transfer on.</li> <li><b>email</b> <i>optional</i> Recipients email address.</li> <li><b>description</b> <i>optional</i> Arbitrary string to describe.</li> <li><b>metadata</b> <i>optional</i> key/value for storing information.</li> <li><b>default_card</b> <i>optional</i> The default card to sue.</li> </ul>

**Value**

A data frame with recipient information

---

stripe\_update\_refund *Update a Refund*

---

**Description**

Update a refund with any metadata you want to include about the refund.

**Usage**

```
stripe_update_refund(api_key, charge_id, refund_id, args)
```

**Arguments**

api_key	Your Stripe API Key
charge_id	The id of the charge with the refund.
refund_id	The id of the refund you want to update
args	A list can contain: <ul style="list-style-type: none"> <li><b>metadata</b> <i>optional</i> Any metadata in a list to change for refund.</li> </ul>

**Value**

A data frame with the updated refund info if successful.

---

```
stripe_update_subscription
```

*Update a subscription.*

---

### Description

Update the information for a subscription on a customer.

### Usage

```
stripe_update_subscription(api_key, customer_id, subscription_id, args)
```

### Arguments

<code>api_key</code>	Your Stripe API Key
<code>customer_id</code>	The id of the customer with the subscription.
<code>subscription_id</code>	The id of the subscription you want to update.
<code>args</code>	which includes: <ul style="list-style-type: none"> <li><b>plan</b> <b>required</b> The plan id</li> <li><b>coupon</b> <i>optional</i> A coupon to apply to this subscription.</li> <li><b>prorate</b> <i>optional</i> Flag whether to prorate plan when switched. defaults: true.</li> <li><b>trial_end</b> <i>optional</i> When the trial ends. Unix timestamp to override plan. You can use <b>now</b> to make it end now.</li> <li><b>card</b> <i>optional</i> A token or card dictionary to use instead of default. <b>required</b> if customer has no card.</li> <li><b>quantity</b> <i>optional</i> Quantity to place in subscription.</li> <li><b>application_fee_percent</b> <i>optional</i> A decimal 1 to 100 for percentage of amount.</li> <li><b>metadata</b> <i>optional</i> A list which can contain anything to further describe.</li> </ul>

### Value

A data frame with the updated subscription info if successful.

---

```
stripe_update_transfer
```

*Update a Transfer*

---

### Description

You can update a transfer that is scheduled or has been accepted by changing description.

**Usage**

```
stripe_update_transfer(api_key, transfer_id, args)
```

**Arguments**

<code>api_key</code>	Your Stripe API Key
<code>transfer_id</code>	The id of the transfer you want to update
<code>args</code>	A list can contain <b>metadata</b> <i>optional</i> A list of metadata to include <b>description</b> <i>optional</i> A string to describe the transfer

**Value**

A data frame with the updated transfer info if successful.

# Index

[rstripe](#), 3  
[rstripe-package \(rstripe\)](#), 3

[stripe\\_balance\\_history](#), 3  
[stripe\\_cancel\\_transfer](#), 4  
[stripe\\_capture\\_charge](#), 4  
[stripe\\_close\\_dispute](#), 5  
[stripe\\_create\\_account\\_token](#), 5  
[stripe\\_create\\_application\\_fee\\_refund](#), 6  
[stripe\\_create\\_card](#), 7  
[stripe\\_create\\_card\\_token](#), 7  
[stripe\\_create\\_charge](#), 8  
[stripe\\_create\\_coupon](#), 9  
[stripe\\_create\\_customer](#), 10  
[stripe\\_create\\_invoice](#), 11  
[stripe\\_create\\_invoice\\_item](#), 11  
[stripe\\_create\\_plan](#), 12  
[stripe\\_create\\_recipient](#), 13  
[stripe\\_create\\_refund](#), 14  
[stripe\\_create\\_subscription](#), 14  
[stripe\\_create\\_transfer](#), 15  
[stripe\\_delete\\_card](#), 16  
[stripe\\_delete\\_coupon](#), 16  
[stripe\\_delete\\_customer](#), 17  
[stripe\\_delete\\_customer\\_discount](#), 17  
[stripe\\_delete\\_invoice\\_item](#), 18  
[stripe\\_delete\\_plan](#), 18  
[stripe\\_delete\\_recipient](#), 19  
[stripe\\_delete\\_subscription](#), 19  
[stripe\\_delete\\_subscription\\_discount](#), 20  
[stripe\\_list\\_application\\_fee\\_refunds](#), 21  
[stripe\\_list\\_application\\_fees](#), 20  
[stripe\\_list\\_cards](#), 21  
[stripe\\_list\\_charges](#), 22  
[stripe\\_list\\_coupons](#), 23  
[stripe\\_list\\_customers](#), 23  
[stripe\\_list\\_events](#), 24  
[stripe\\_list\\_invoice\\_items](#), 25  
[stripe\\_list\\_invoices](#), 25  
[stripe\\_list\\_plans](#), 26  
[stripe\\_list\\_recipients](#), 26  
[stripe\\_list\\_refunds](#), 27  
[stripe\\_list\\_subscriptions](#), 27  
[stripe\\_list\\_transfers](#), 28  
[stripe\\_pay\\_invoice](#), 28  
[stripe\\_retrieve\\_account](#), 29  
[stripe\\_retrieve\\_application\\_fee](#), 29  
[stripe\\_retrieve\\_application\\_fee\\_refund](#), 30  
[stripe\\_retrieve\\_balance](#), 30  
[stripe\\_retrieve\\_balance\\_transaction](#), 31  
[stripe\\_retrieve\\_card](#), 31  
[stripe\\_retrieve\\_charge](#), 32  
[stripe\\_retrieve\\_coupon](#), 32  
[stripe\\_retrieve\\_customer](#), 33  
[stripe\\_retrieve\\_event](#), 33  
[stripe\\_retrieve\\_invoice](#), 34  
[stripe\\_retrieve\\_invoice\\_item](#), 34  
[stripe\\_retrieve\\_invoice\\_line\\_items](#), 35  
[stripe\\_retrieve\\_plan](#), 35  
[stripe\\_retrieve\\_recipient](#), 36  
[stripe\\_retrieve\\_refund](#), 36  
[stripe\\_retrieve\\_subscription](#), 37  
[stripe\\_retrieve\\_transfer](#), 37  
[stripe\\_upcoming\\_customer\\_invoice](#), 38  
[stripe\\_update\\_application\\_fee\\_refund](#), 38  
[stripe\\_update\\_card](#), 39  
[stripe\\_update\\_charge](#), 40  
[stripe\\_update\\_coupon](#), 40  
[stripe\\_update\\_customer](#), 41  
[stripe\\_update\\_dispute](#), 42  
[stripe\\_update\\_invoice](#), 42  
[stripe\\_update\\_invoice\\_item](#), 43  
[stripe\\_update\\_plan](#), 44

stripe\_update\_recipient, [44](#)  
stripe\_update\_refund, [45](#)  
stripe\_update\_subscription, [46](#)  
stripe\_update\_transfer, [46](#)