Robot Framework

RequestsLibrary

Sample Test Cases with Shopping Cart

- Checkout Product from Store Service flow

1. Get Product List
2. Get Product Detail
3. Order Product
4. Confirm Payment

Sample Test Cases with Shopping Cart

- Test Cases: Checkout Dinner Set from Store Service

1. Get Product List
    - input URI PATH: /api/v1/product
    - input HTTP Method: GET
    - input HTTP Headers: Accept application/json
    - assert http status code should be 200
    - assert products should have 2 items
    - assert products should contain '43 Piece dinner Set'

2. Get Product Detail
    - input URI PATH: /api/v1/product/{product id}
    - input HTTP Method: GET
    - input HTTP Headers: Accept application/json
    - assert http status code should be 200
    - assert product name should contain '43 Piece dinner Set'

Sample Test Cases with Shopping Cart

- Test Cases: Checkout Dinner Set from Store Service

3.Order Product

- ○ input URI PATH: /api/v1/order
- ○ input HTTP Method: POST
- ○ input HTTP Headers: Accept application/json
- ○ input HTTP Headers: Content-Type application/json
- ○ input HTTP Body
- ○ assert http status code should be 200
- ○ assert product total price should be 14.95

4.Confirm Payment

- ○ input URI PATH: /api/v1/confirmPayment
- ○ input HTTP Method: POST
- ○ input HTTP Headers: Accept application/json
- ○ input HTTP Headers: Content-Type application/json
- ○ input HTTP Body
- ○ assert http status code should be 200
- ○ assert notification message should be 'วันเวลาที่ชำระเงิน 1/3/2020 13:30:00 หมายเลขคำสั่งซื้อ 8004359104 คุณสามารถติดตามสินค้าผ่านช่องทาง Kerry หมายเลข 1785261900'

Sample Test Cases with Shopping Cart

PATH: /api/v1/product
Method: GET
Headers:
  Accept application/json

Get Product List

http status code is 200
products have 2 items

PATH: /api/v1/product/{product id}
Method: GET
Headers:
  Accept application/json

Get Product Detail

http status code is 200
product name is '43 Piece dinner Set'

PATH: /api/v1/order
Method: POST
Headers:
  Accept application/json
  Content-Type application/json
Body

Order

http status code is 200
product total price is 14.95
order id is 8004359104

PATH: /api/v1/confirmPayment
Method: POST
Headers:
  Accept application/json
  Content-Type application/json
Body

Confirm Payment

notification message is 'วันเวลาที่ชำระเงิน 1/3/2020 13:30:00 หมายเลขคำสั่งซื้อ 8004359104 คุณสามารถติดตามสินค้าผ่านช่องทาง Kerry หมายเลข 1785261900'

© 2020 Siam Chamnankit Company Limited

Sample Test Cases with Shopping Cart

PATH: /api/v1/product

Method: GET

Headers:

Accept application/json

Get Product List

http status code is 200

products have 2 items

PATH: /api/v1/product/{product id}

Method: GET

Headers:

Accept application/json

Get Product Detail

http status code is 200

product name is '43 Piece dinner Set'

PATH: /api/v1/order

Method: POST

Headers:

Accept application/json

Content-Type application/json

Body

Order

http status code is 200

product total price is 14.95

order id is 8004359104

PATH: /api/v1/confirmPayment

Method: POST

Headers:

Accept application/json

Content-Type application/json

Body

Confirm Payment

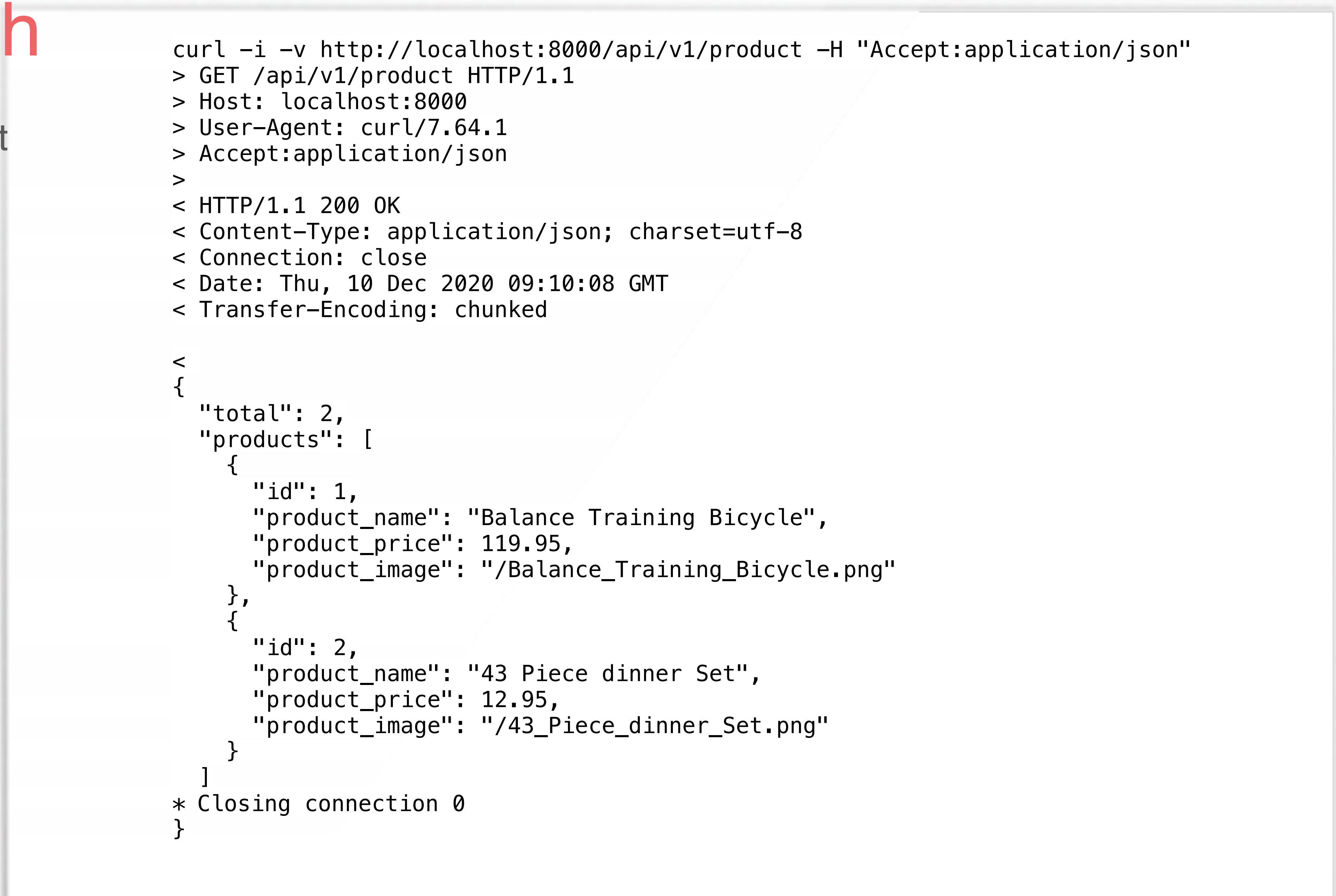
notification message is 'วันเวลาที่ชำระเงิน 1/3/2020 13:30:00 หมายเลขคำสั่งซื้อ 8004359104 คุณสามารถติดตามสินค้าผ่านช่องทาง Kerry หมายเลข 1785261900'

© 2020 Siam Chamnankit Company Limited

HTTP underneath

Call Get Product List

via curl



```
curl -i -v http://localhost:8000/api/v1/product -H "Accept:application/json"
> GET /api/v1/product HTTP/1.1
> Host: localhost:8000
> User-Agent: curl/7.64.1
> Accept:application/json
>
< HTTP/1.1 200 OK
< Content-Type: application/json; charset=utf-8
< Connection: close
< Date: Thu, 10 Dec 2020 09:10:08 GMT
< Transfer-Encoding: chunked

<
{
  "total": 2,
  "products": [
    {
      "id": 1,
      "product_name": "Balance Training Bicycle",
      "product_price": 119.95,
      "product_image": "/Balance_Training_Bicycle.png"
    },
    {
      "id": 2,
      "product_name": "43 Piece dinner Set",
      "product_price": 12.95,
      "product_image": "/43_Piece_dinner_Set.png"
    }
  ]
* Closing connection 0
}
```

Sample Test Cases with Shopping Cart

- How to pass the return order id from 'Order Diner Set' to 'Confirm Payment' order id?
- Which parameters in each task that we have to concern?
  - consequence
  - specific

HTTP RequestsLibrary(Python)

Library for HTTP level testing using Python Requests internally.

lean more

© 2020 Siam Chamnankit Company Limited

Prerequisite(Global Install)

1. Python3

2. Robot Framework
   ```
   pip install robotframework
   ```

3. HTTP RequestsLibrary(Python)
   ```
   pip install robotframework-requests
   ```

4. Check the installed lib
   ```
   pip list
   ```

© 2020 Siam Chamnankit Company Limited

Prerequisite(Local Install): Linux/Mac

1. Python3

2. Create Virtual Env

   ```
   python -m venv env
   source env/bin/activate
   ```

3. Robot Framework

   ```
   pip install robotframework
   ```

4. HTTP RequestsLibrary(Python)

   ```
   pip install robotframework-requests
   ```

5. Check the installed lib

   ```
   pip list
   ```

© 2020 Siam Chamnankit Company Limited

Prerequisite(Local Install): Windows

1. Python3

2. Create Virtual Env

   ```
   python -m venv env
   env\Scripts\activate.bat
   ```

3. Robot Framework

   ```
   pip install robotframework
   ```

4. HTTP RequestsLibrary(Python)

   ```
   pip install robotframework-requests
   ```

5. Check the installed lib

   ```
   pip list
   ```

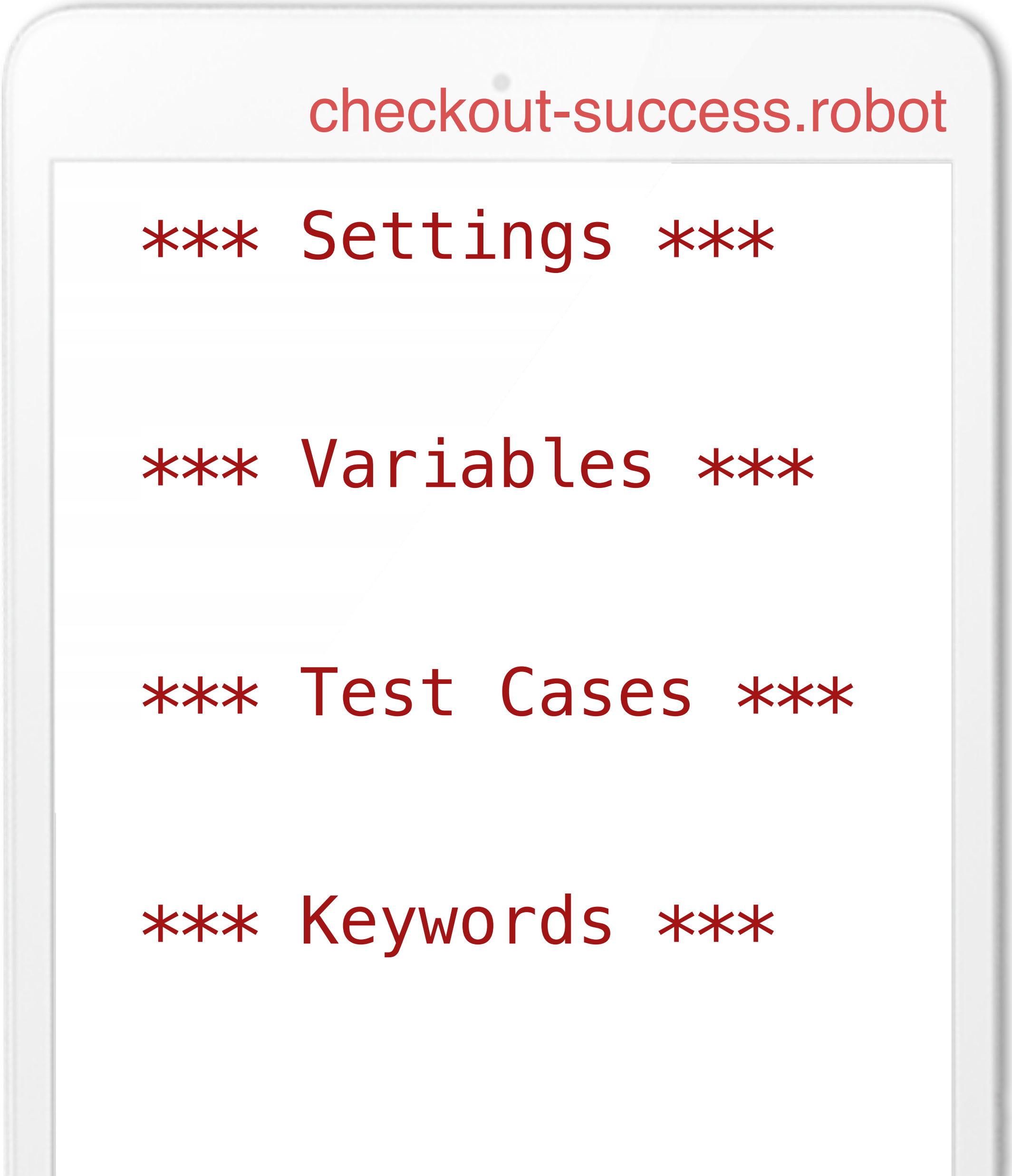
© 2020 Siam Chamnankit Company Limited

Sections in Test Case Files

Robot Framework data is defined in different sections.

lean more

```
Create Directory
call request-json
```

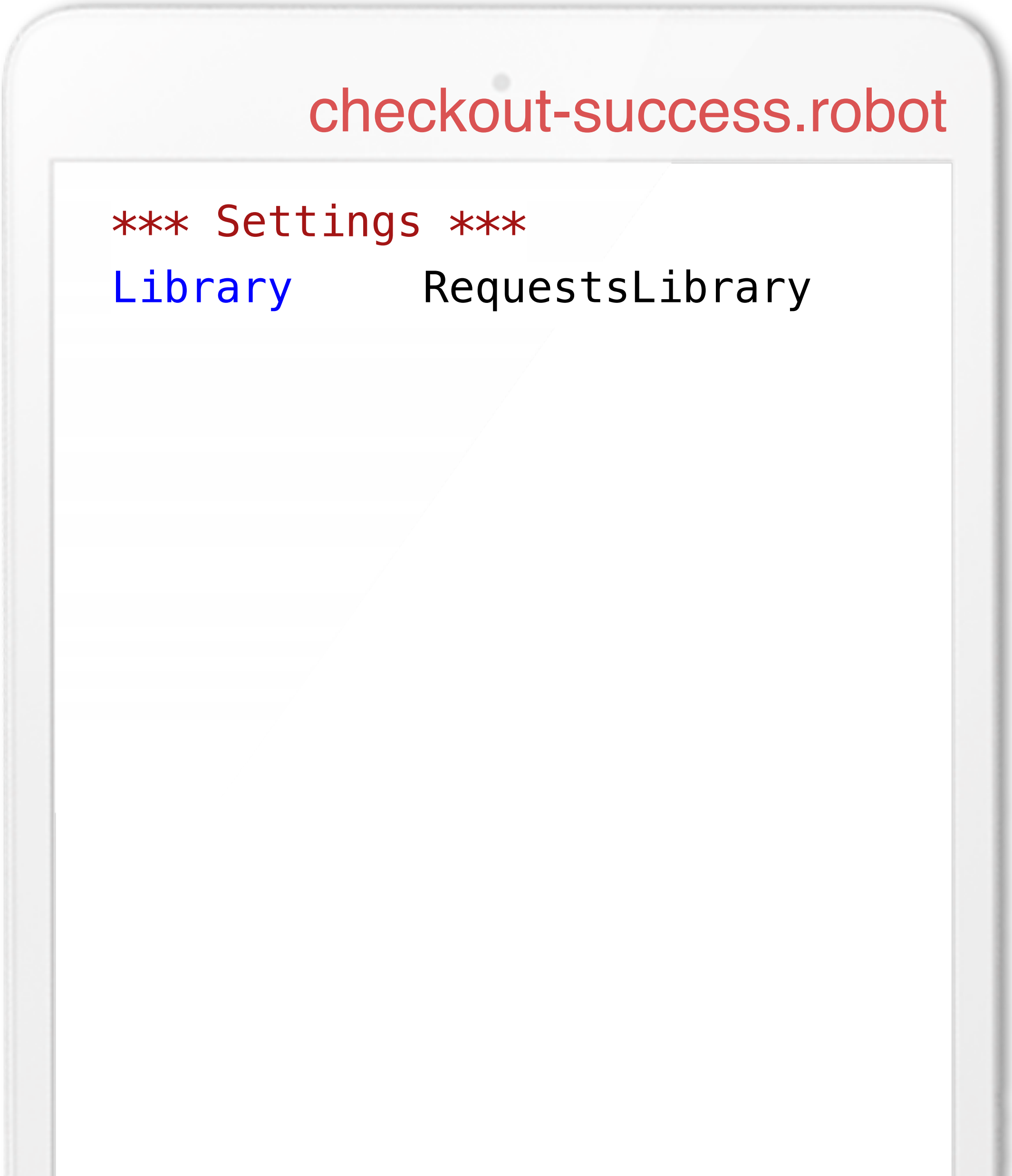


checkout-success.robot

```
*** Settings ***


*** Variables ***


*** Test Cases ***


*** Keywords ***
```

Settings
in Test Case Files

import RequestsLibrary

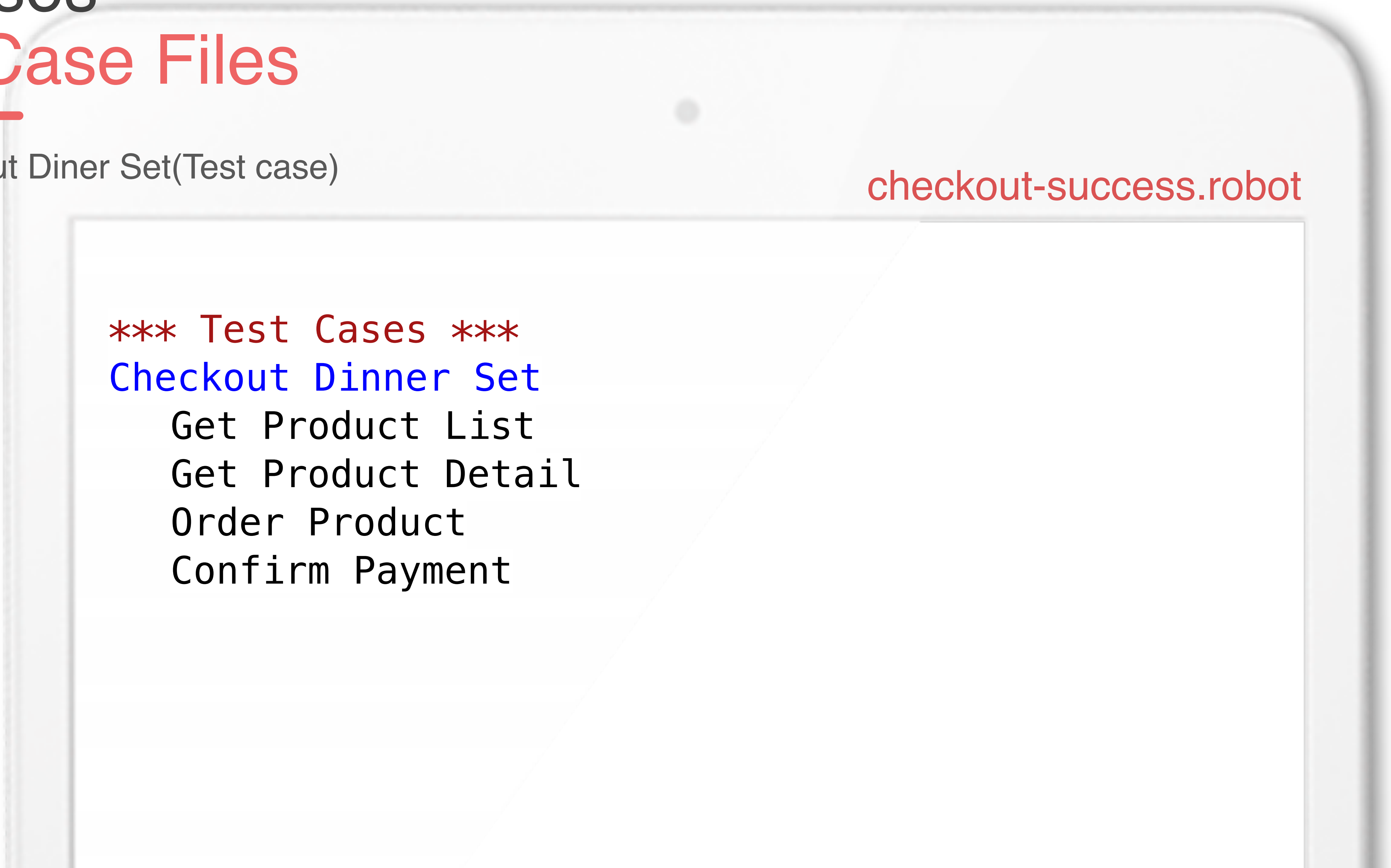


checkout-success.robot

```
*** Settings ***
Library      RequestsLibrary
```

Test Cases
in Test Case Files

Create Checkout Diner Set(Test case)

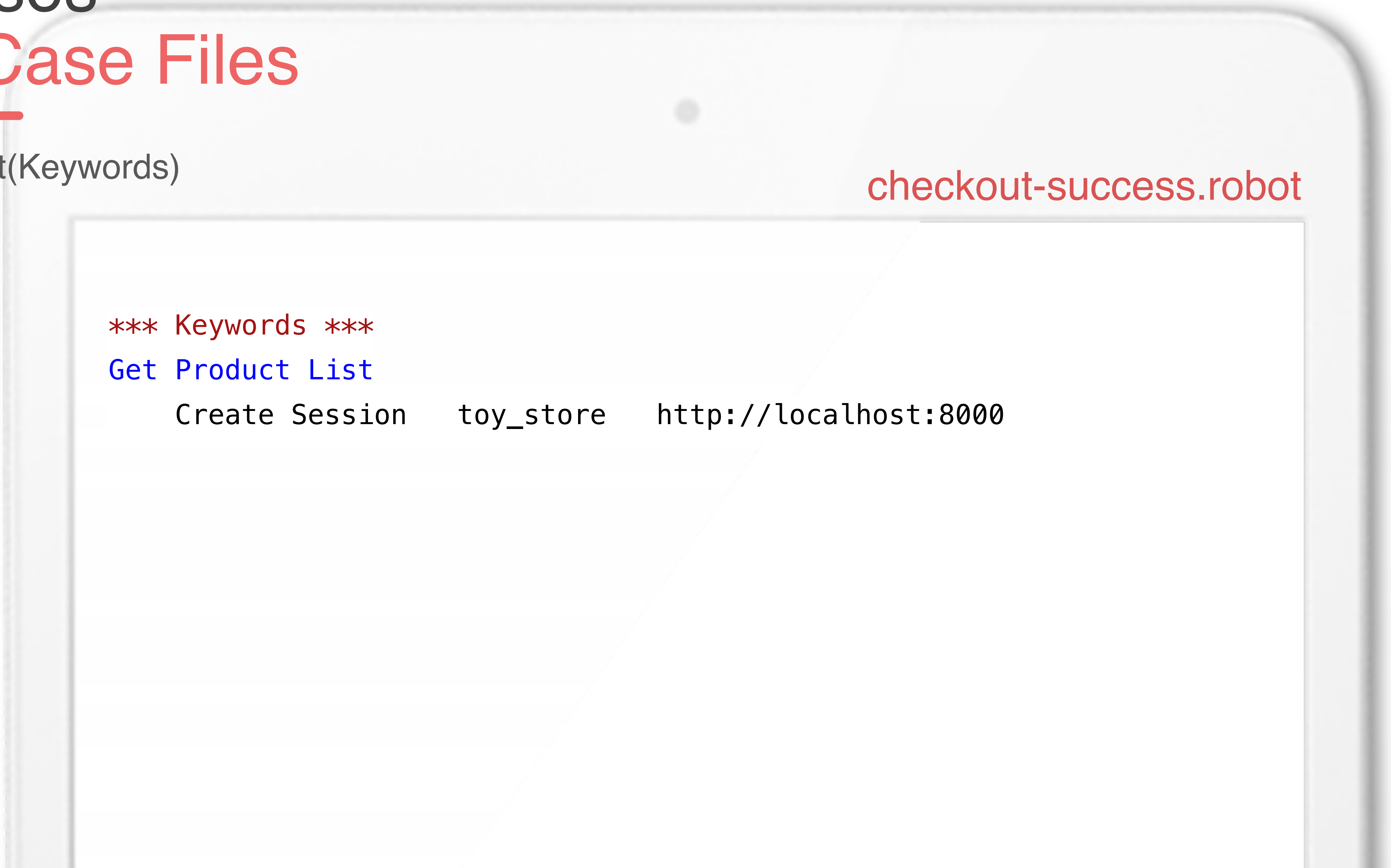


checkout-success.robot

```
*** Test Cases ***
Checkout Dinner Set
    Get Product List
    Get Product Detail
    Order Product
    Confirm Payment
```

Test Cases in Test Case Files

Get Product List(Keywords)

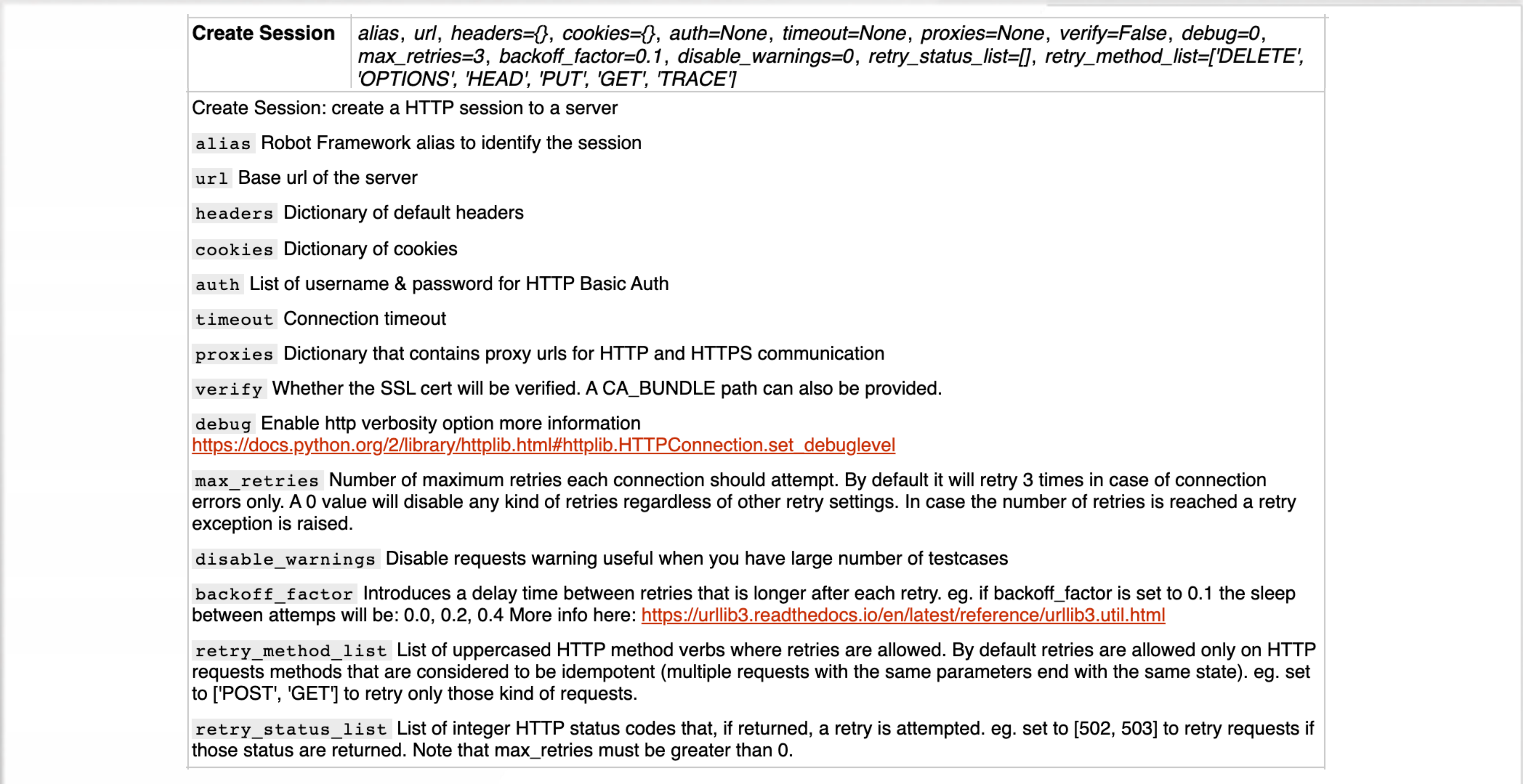


checkout-success.robot

```
*** Keywords ***
Get Product List
    Create Session    toy_store    http://localhost:8000
```

RequestsLibrary Create Session API

lean more



Create Session	*alias, url, headers={}, cookies={}, auth=None, timeout=None, proxies=None, verify=False, debug=0, max_retries=3, backoff_factor=0.1, disable_warnings=0, retry_status_list=[], retry_method_list=['DELETE', 'OPTIONS', 'HEAD', 'PUT', 'GET', 'TRACE']*

Create Session: create a HTTP session to a server

`alias` Robot Framework alias to identify the session

`url` Base url of the server

`headers` Dictionary of default headers

`cookies` Dictionary of cookies

`auth` List of username & password for HTTP Basic Auth

`timeout` Connection timeout

`proxies` Dictionary that contains proxy urls for HTTP and HTTPS communication

`verify` Whether the SSL cert will be verified. A CA_BUNDLE path can also be provided.

`debug` Enable http verbosity option more information https://docs.python.org/2/library/httplib.html#httplib.HTTPConnection.set_debuglevel

`max_retries` Number of maximum retries each connection should attempt. By default it will retry 3 times in case of connection errors only. A 0 value will disable any kind of retries regardless of other retry settings. In case the number of retries is reached a retry exception is raised.

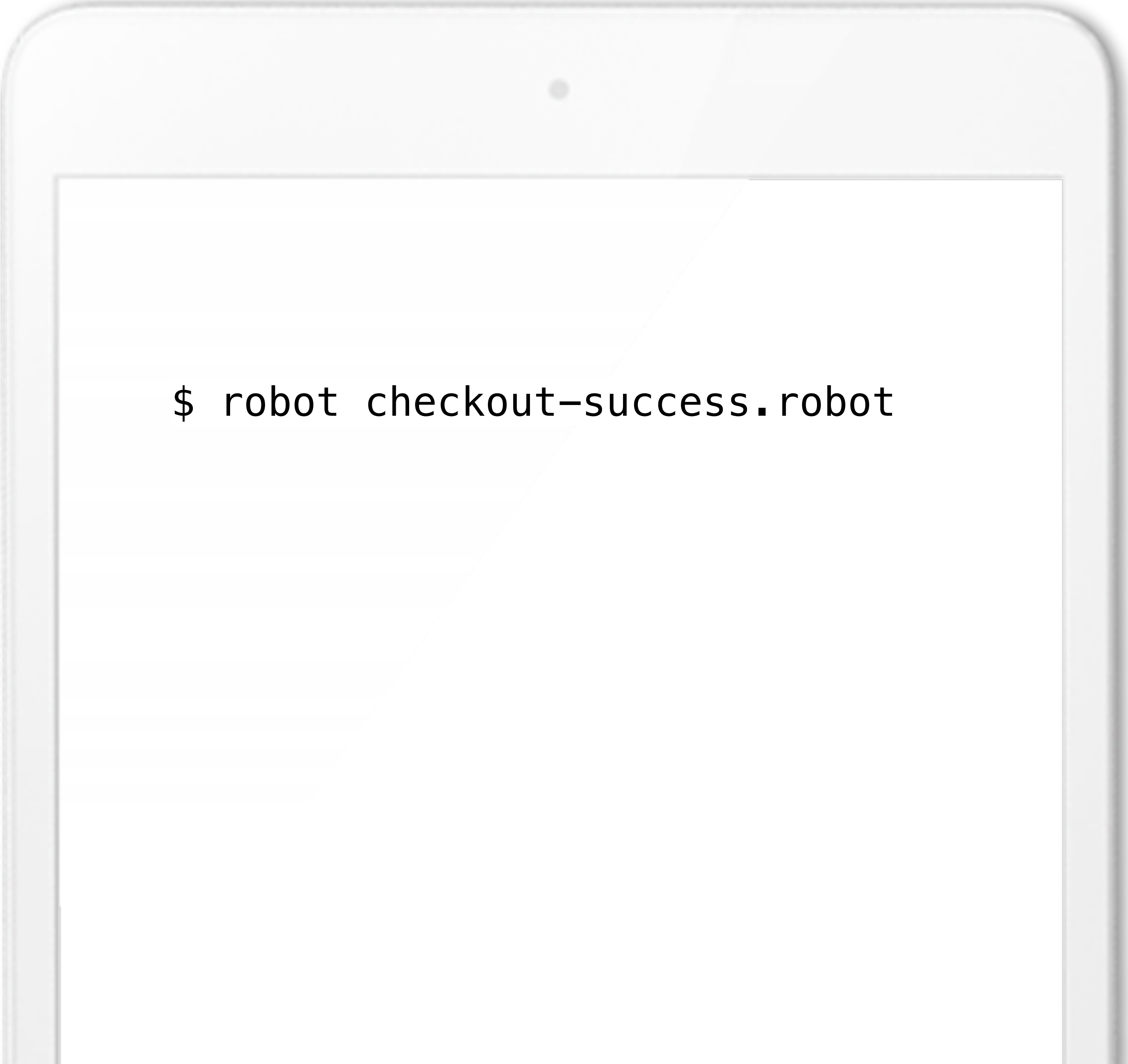
`disable_warnings` Disable requests warning useful when you have large number of testcases

`backoff_factor` Introduces a delay time between retries that is longer after each retry. eg. if backoff_factor is set to 0.1 the sleep between attemps will be: 0.0, 0.2, 0.4 More info here: https://urllib3.readthedocs.io/en/latest/reference/urllib3.util.html

`retry_method_list` List of uppercased HTTP method verbs where retries are allowed. By default retries are allowed only on HTTP requests methods that are considered to be idempotent (multiple requests with the same parameters end with the same state). eg. set to ['POST', 'GET'] to retry only those kind of requests.

`retry_status_list` List of integer HTTP status codes that, if returned, a retry is attempted. eg. set to [502, 503] to retry requests if those status are returned. Note that max_retries must be greater than 0.

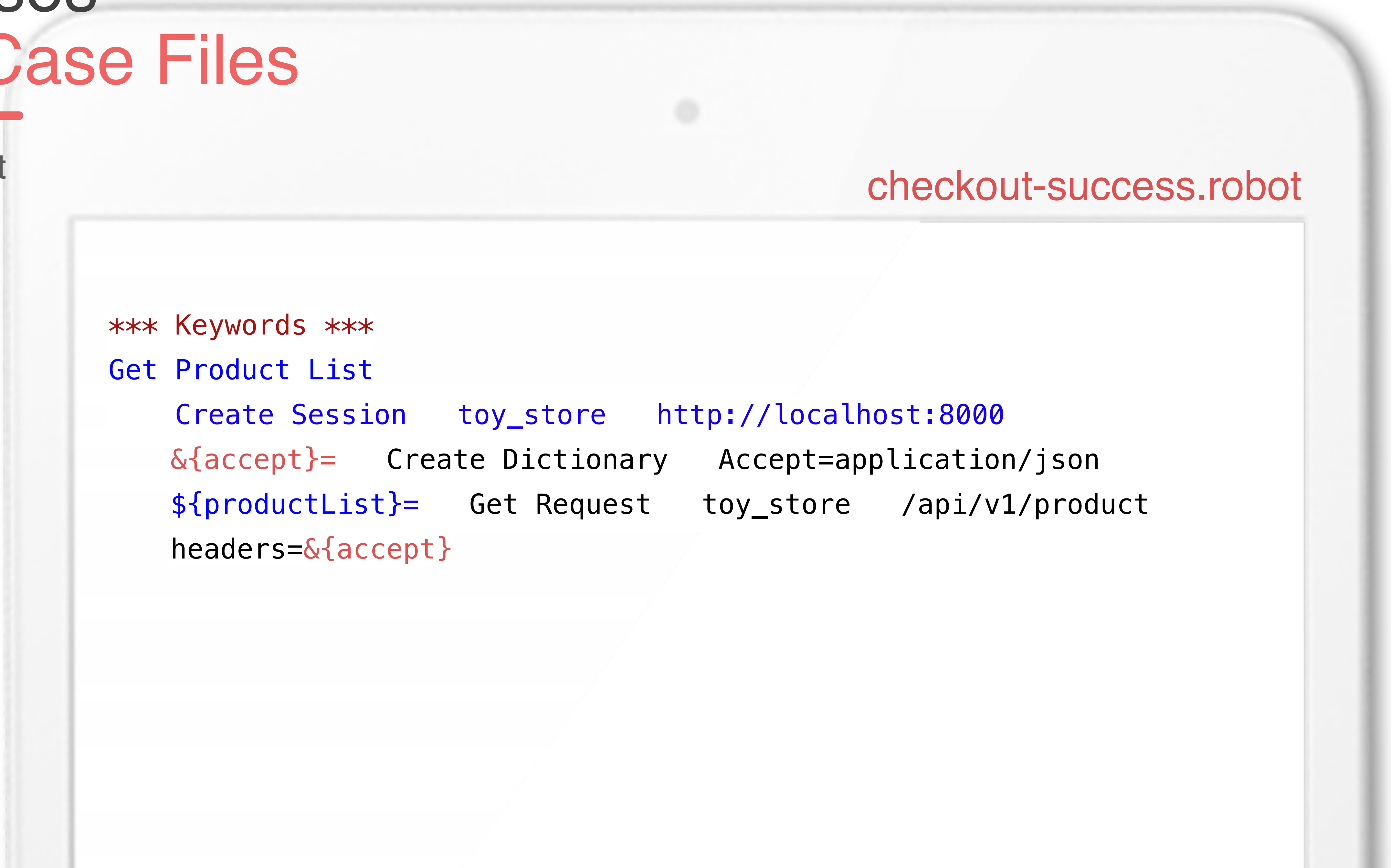
Run
Test Case Files



```
$ robot checkout-success.robot
```

Test Cases in Test Case Files

Get Product List

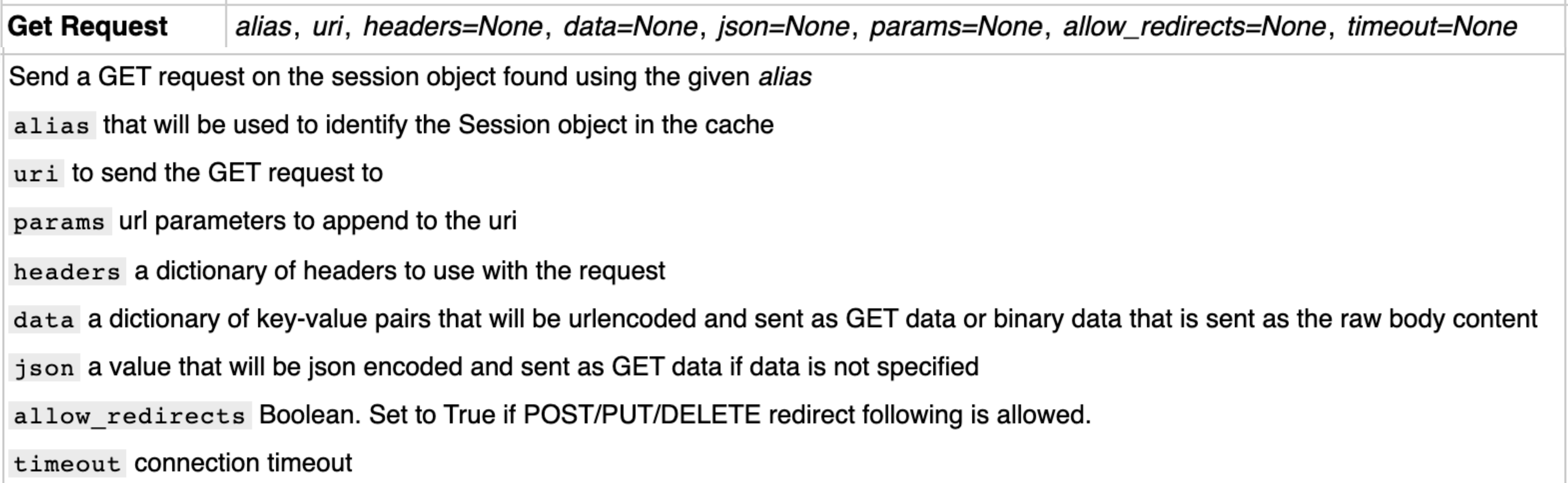


checkout-success.robot

```
*** Keywords ***
Get Product List
    Create Session   toy_store   http://localhost:8000
    &{accept}=   Create Dictionary   Accept=application/json
    ${productList}=   Get Request   toy_store   /api/v1/product
    headers=&{accept}
```

RequestsLibrary
Get Request API

lean more



Get Request	*alias*, *uri*, *headers=None*, *data=None*, *json=None*, *params=None*, *allow_redirects=None*, *timeout=None*

Send a GET request on the session object found using the given *alias*

`alias` that will be used to identify the Session object in the cache

`uri` to send the GET request to

`params` url parameters to append to the uri

`headers` a dictionary of headers to use with the request

`data` a dictionary of key-value pairs that will be urlencoded and sent as GET data or binary data that is sent as the raw body content

`json` a value that will be json encoded and sent as GET data if data is not specified

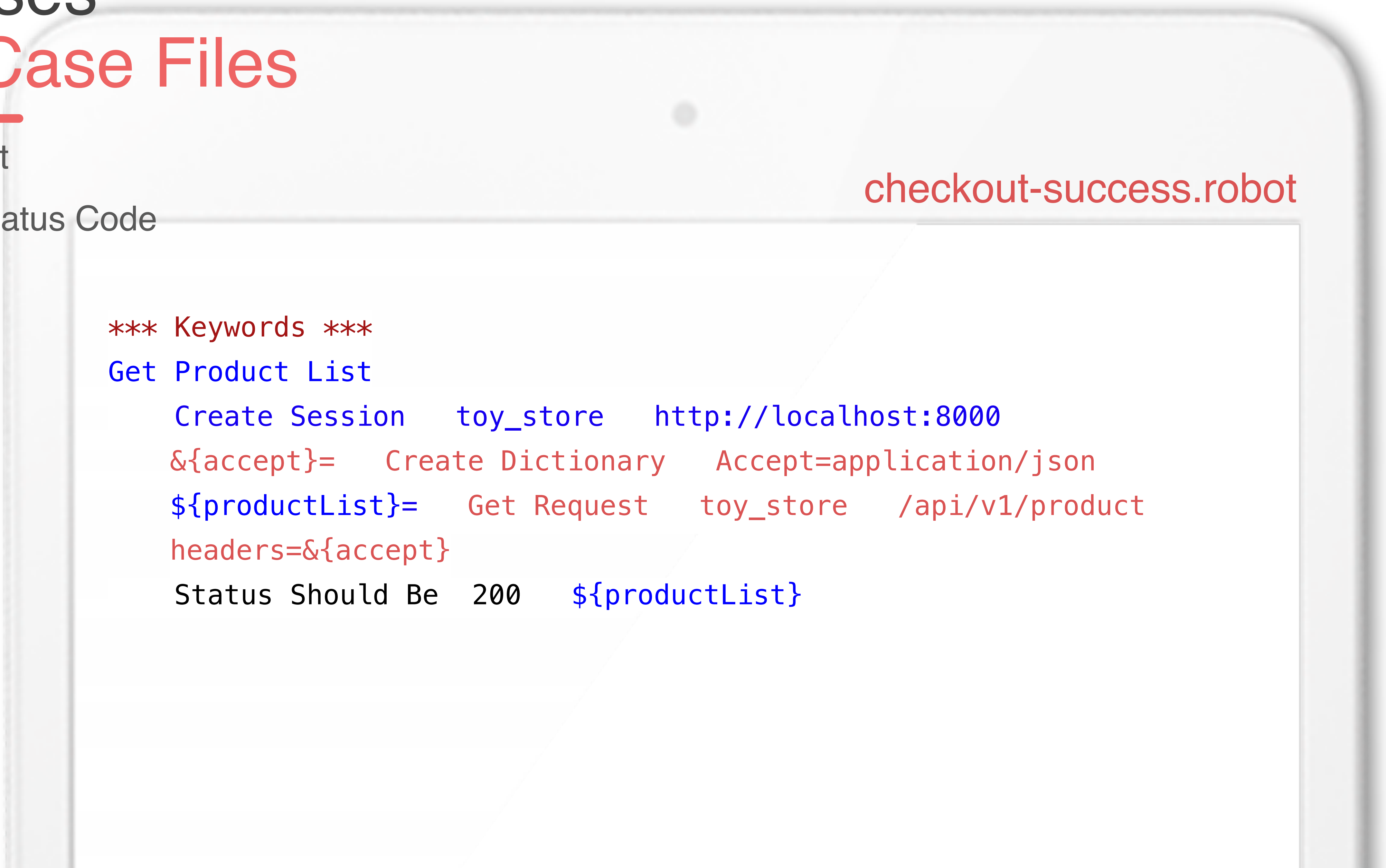
`allow_redirects` Boolean. Set to True if POST/PUT/DELETE redirect following is allowed.

`timeout` connection timeout

Test Cases
in Test Case Files

Get Product List

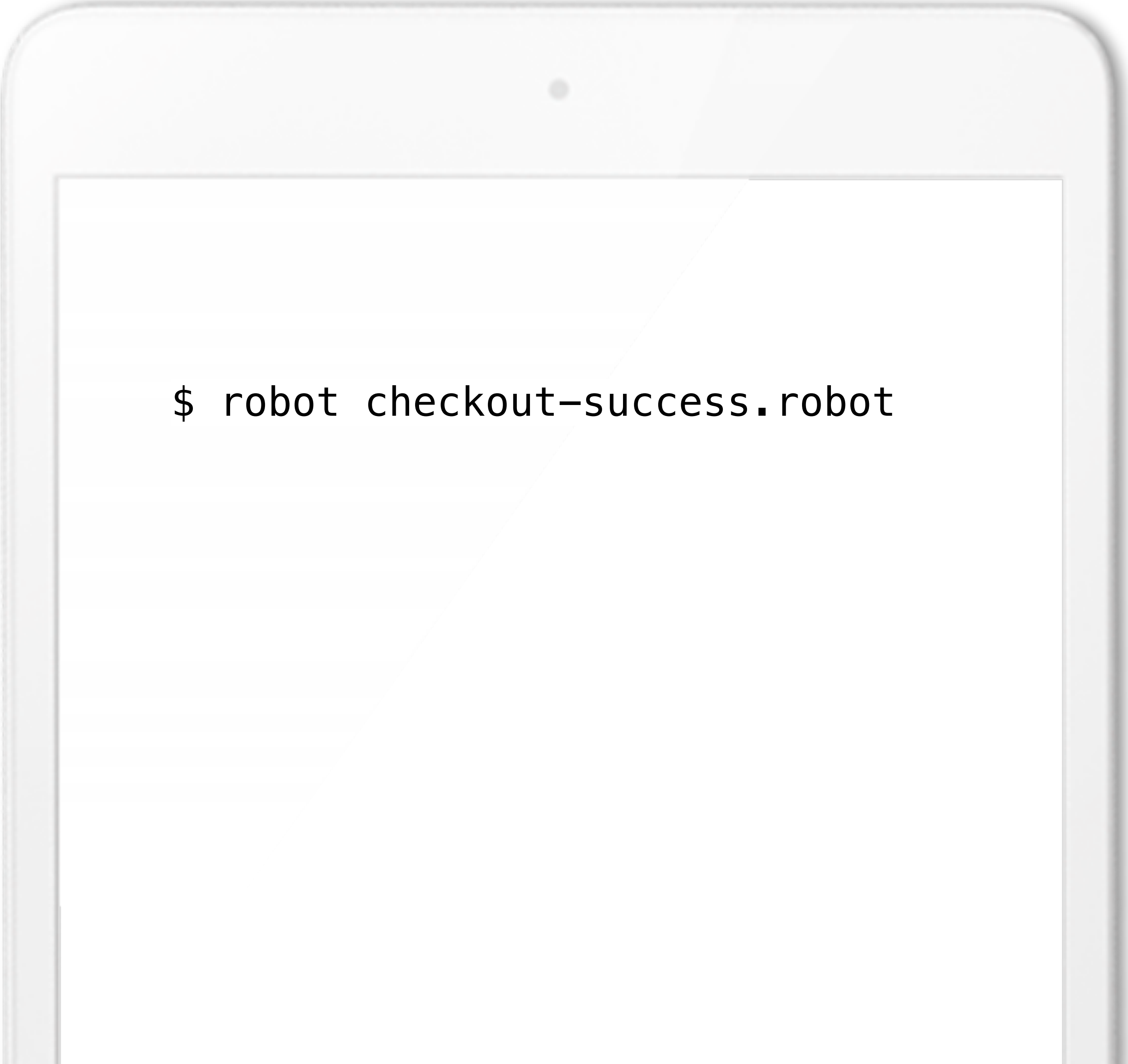
Assert HTTP Status Code



checkout-success.robot

```
*** Keywords ***
Get Product List
    Create Session   toy_store   http://localhost:8000
    &{accept}=   Create Dictionary   Accept=application/json
    ${productList}=   Get Request   toy_store   /api/v1/product
    headers=&{accept}
    Status Should Be  200   ${productList}
```

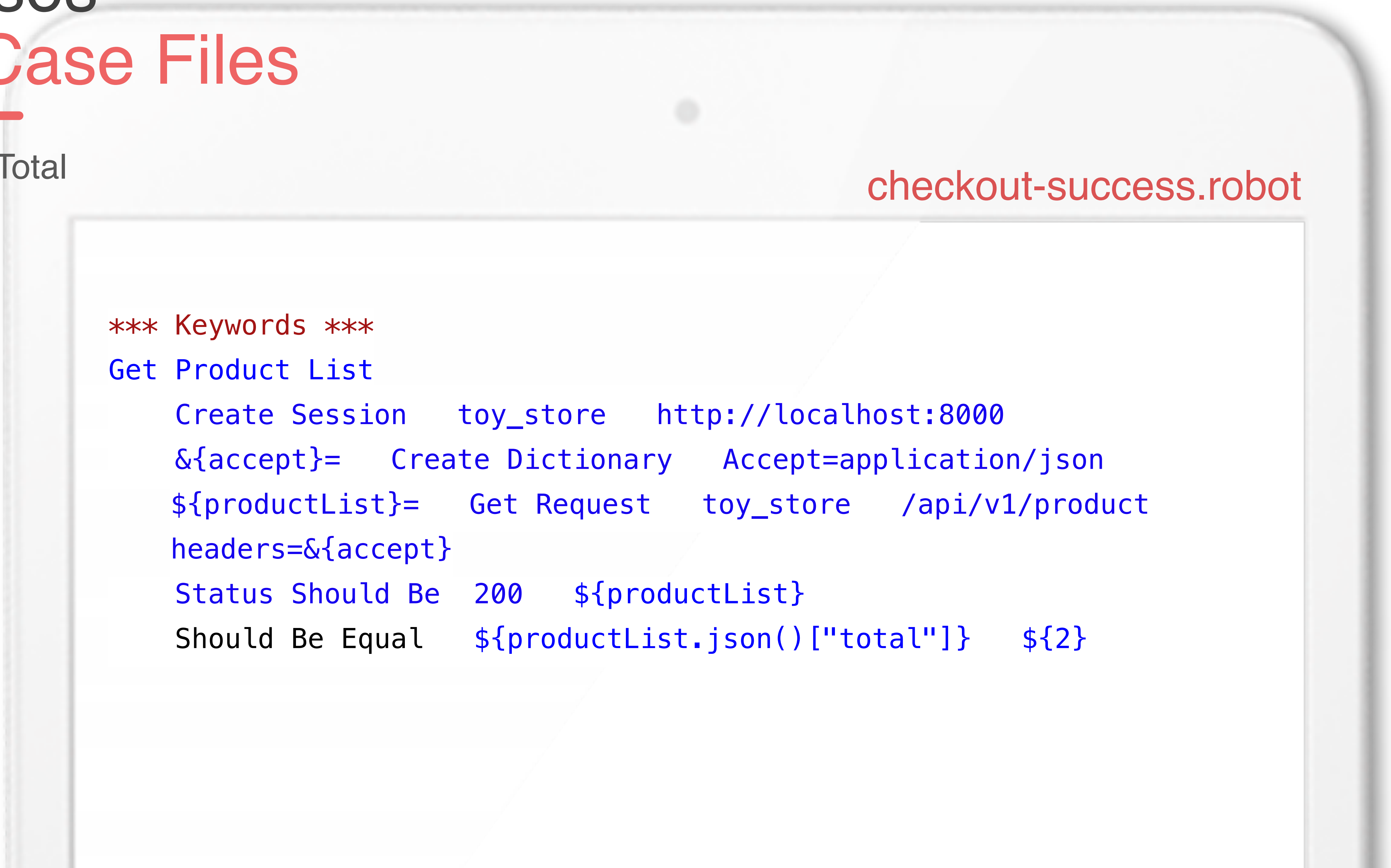
Run
Test Case Files



```
$ robot checkout-success.robot
```

Test Cases
in Test Case Files

Assert Product Total



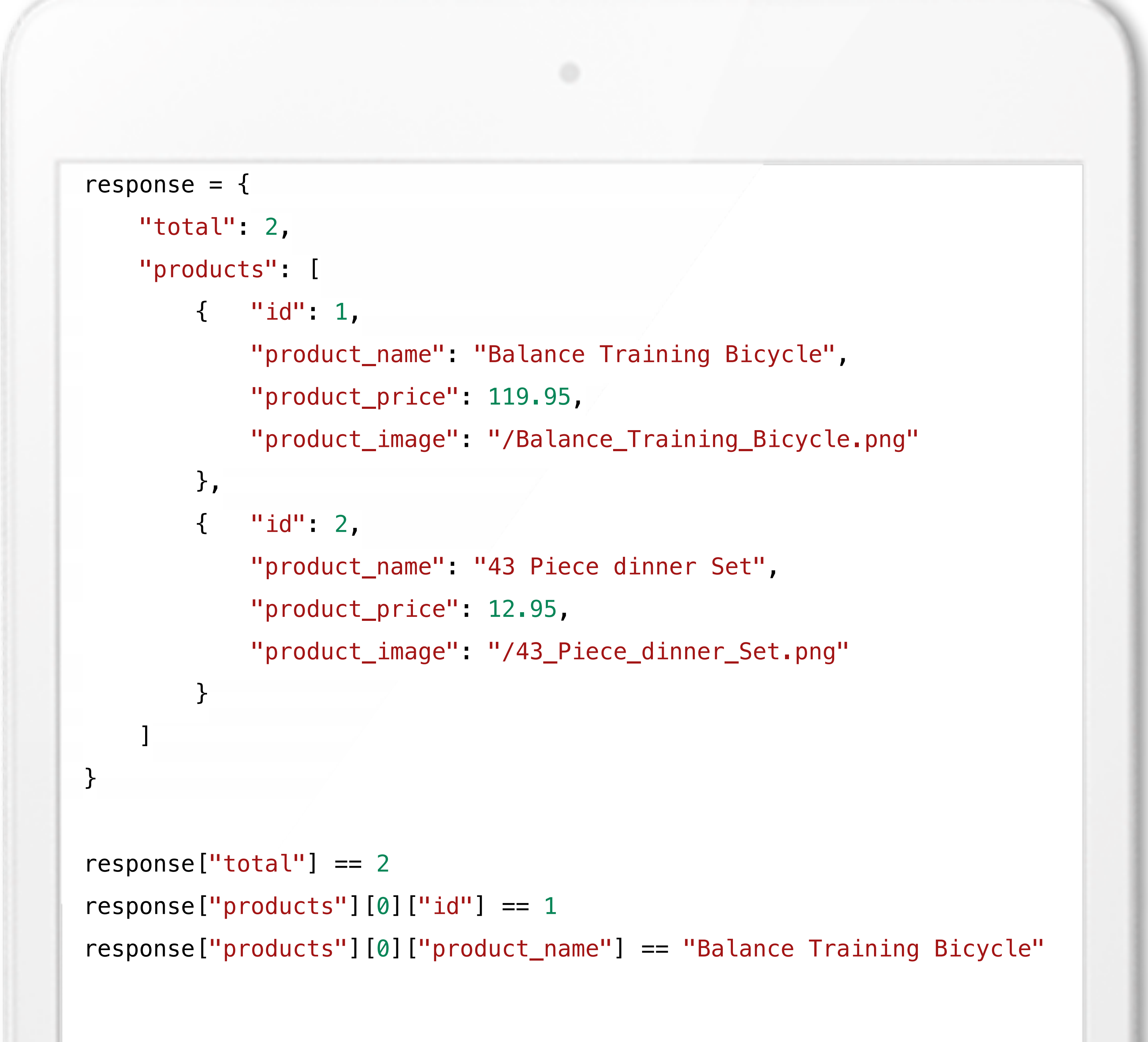
checkout-success.robot

```
*** Keywords ***
Get Product List
    Create Session   toy_store   http://localhost:8000
    &{accept}=   Create Dictionary   Accept=application/json
    ${productList}=   Get Request   toy_store   /api/v1/product
    headers=&{accept}
    Status Should Be  200   ${productList}
    Should Be Equal   ${productList.json()["total"]}   ${2}
```

Variables in Robotframework

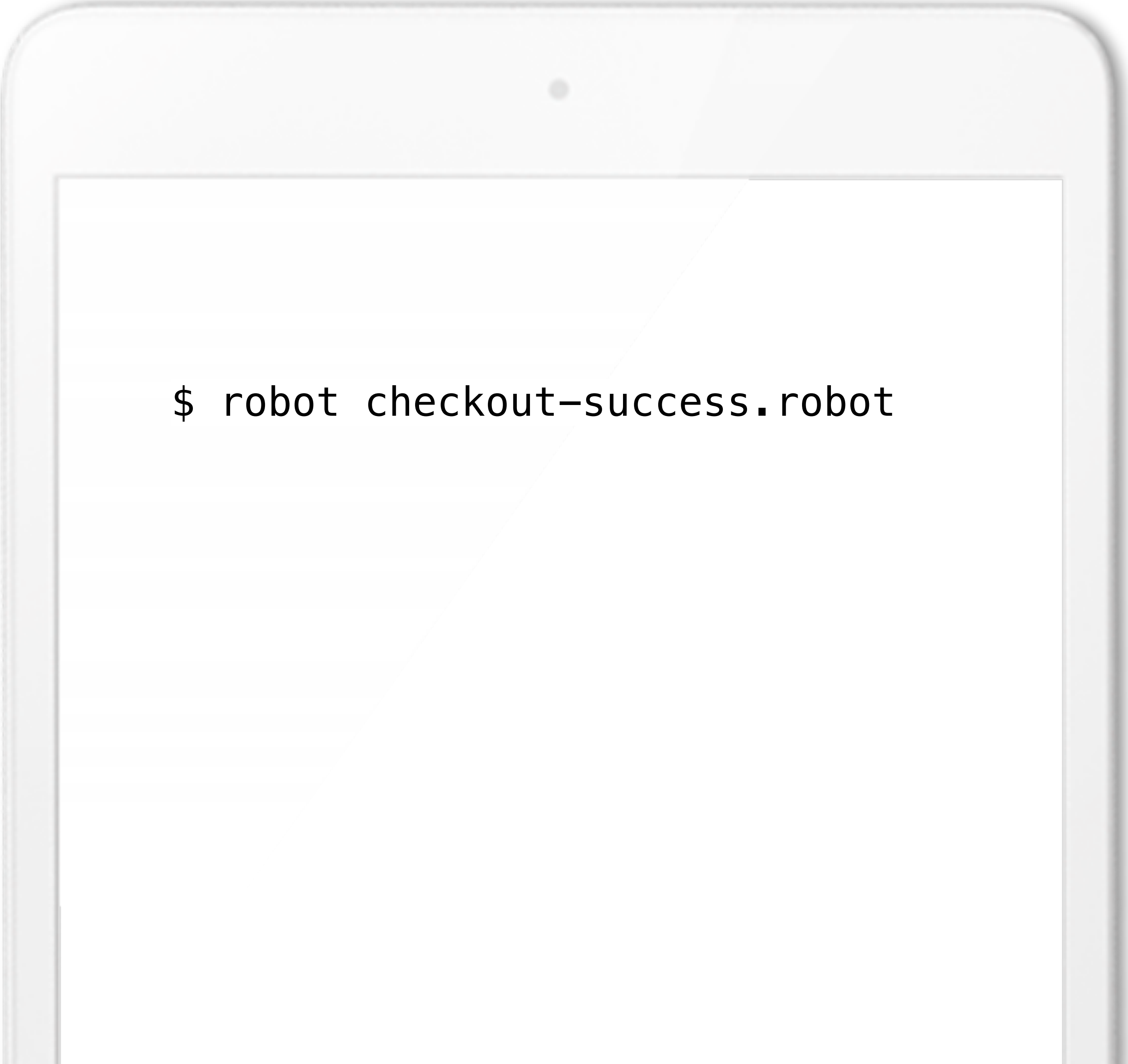
Dictionary

lean more



```
response = {
    "total": 2,
    "products": [
        {   "id": 1,
            "product_name": "Balance Training Bicycle",
            "product_price": 119.95,
            "product_image": "/Balance_Training_Bicycle.png"
        },
        {   "id": 2,
            "product_name": "43 Piece dinner Set",
            "product_price": 12.95,
            "product_image": "/43_Piece_dinner_Set.png"
        }
    ]
}

response["total"] == 2
response["products"][0]["id"] == 1
response["products"][0]["product_name"] == "Balance Training Bicycle"
```

Run
Test Case Files

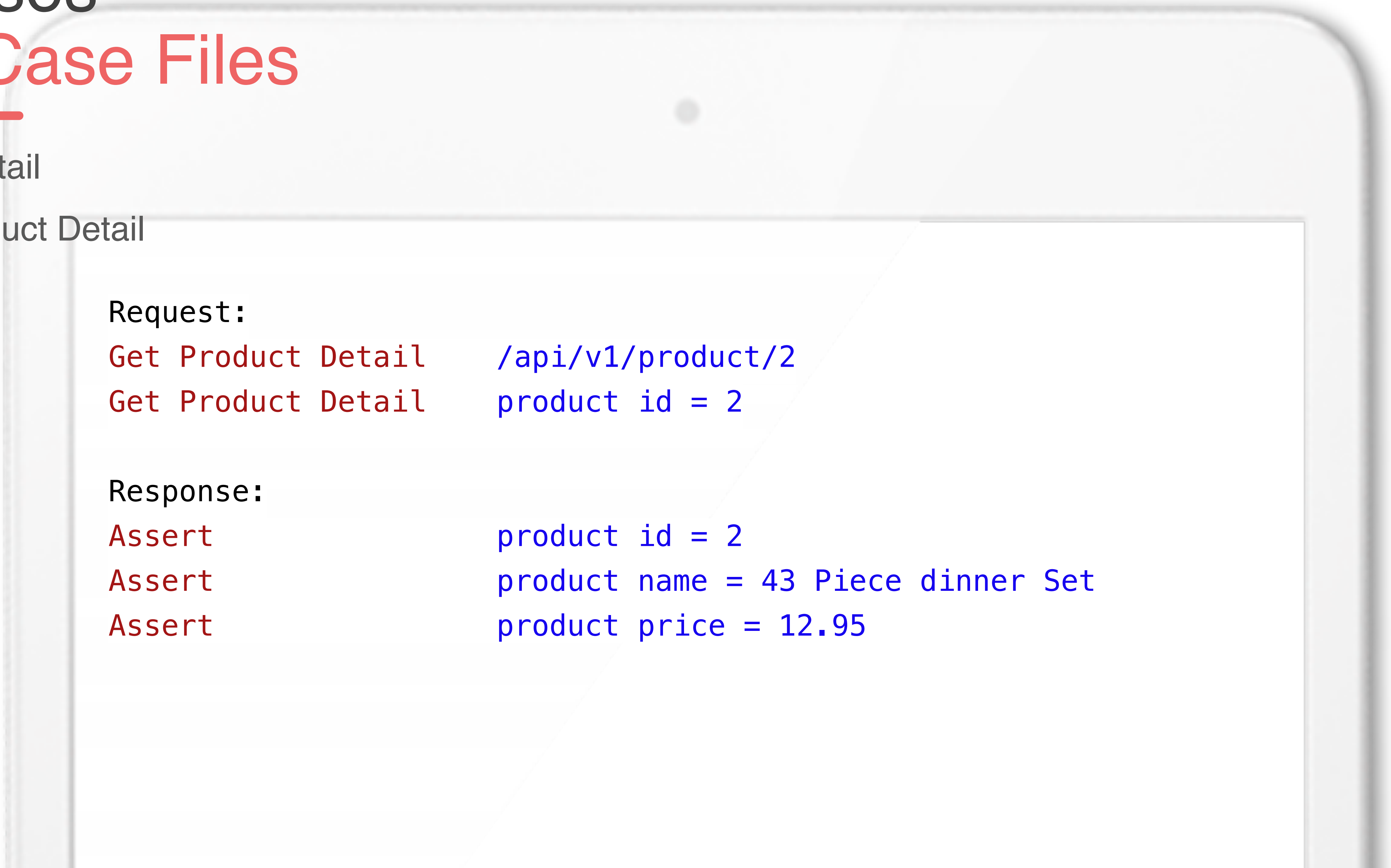


```
$ robot checkout-success.robot
```

Test Cases in Test Case Files

Get Product Detail

Assert Product Detail



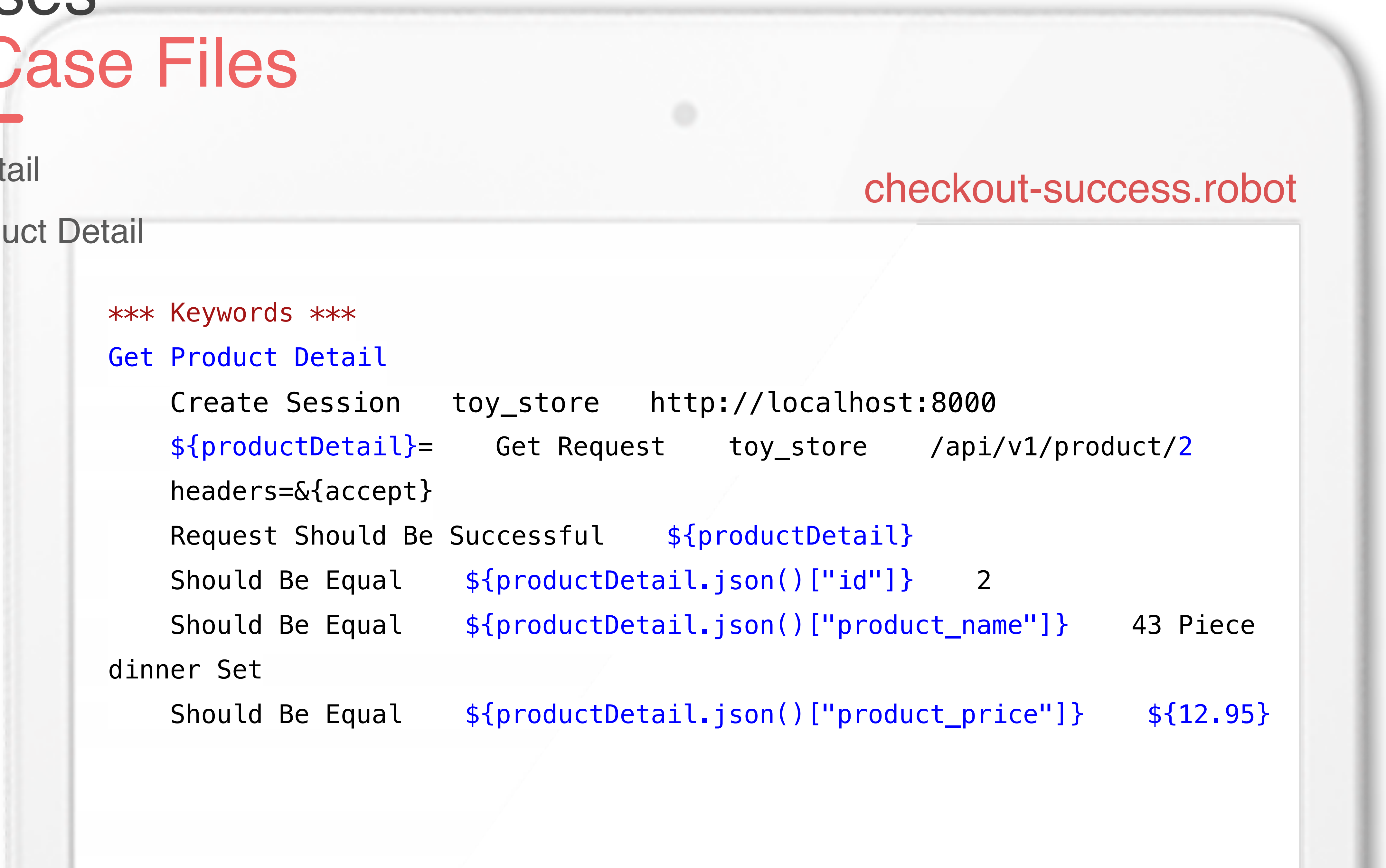
```
Request:
Get Product Detail    /api/v1/product/2
Get Product Detail    product id = 2

Response:
Assert                product id = 2
Assert                product name = 43 Piece dinner Set
Assert                product price = 12.95
```

Test Cases in Test Case Files

Get Product Detail

    Assert Product Detail

checkout-success.robot

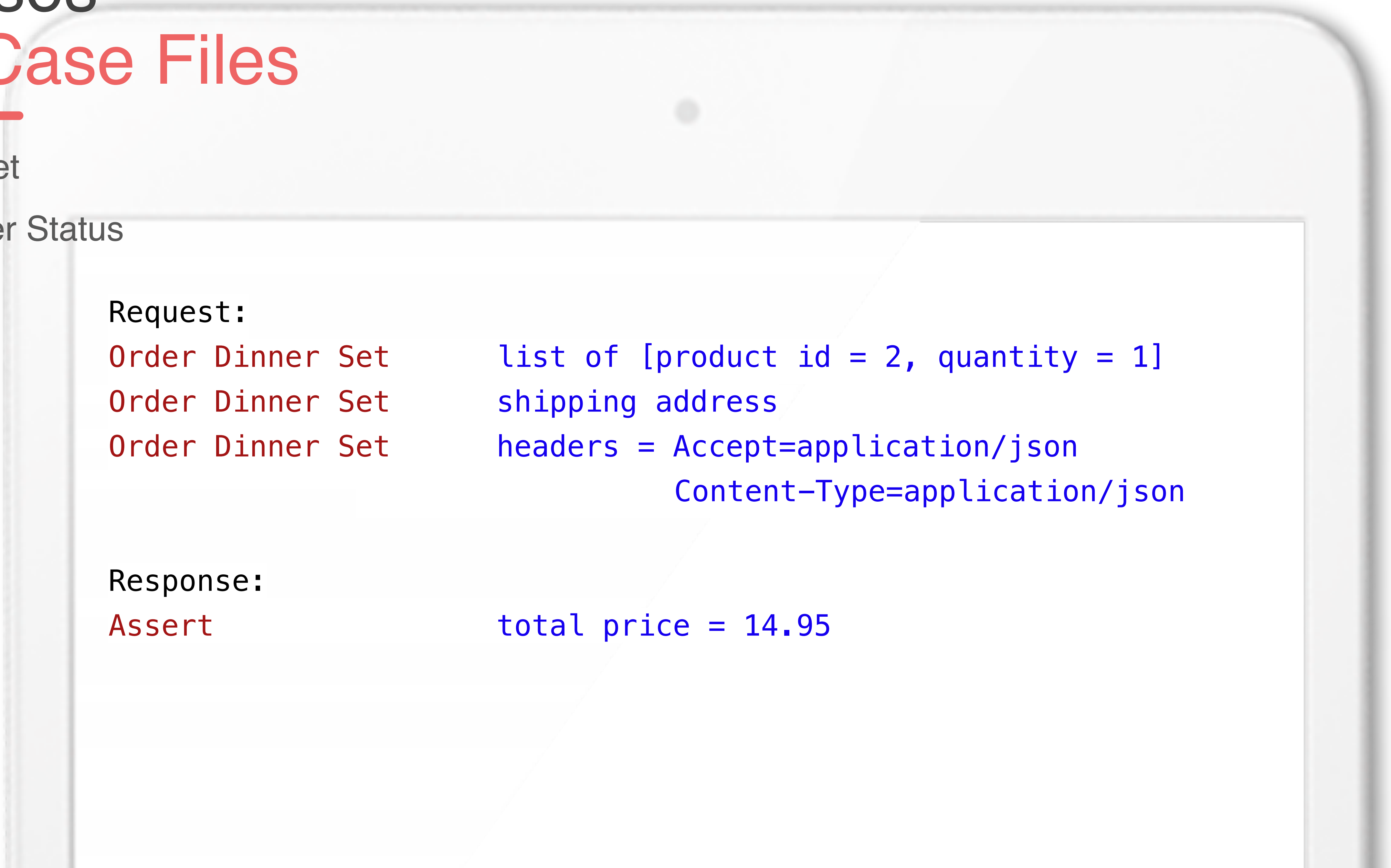


```
*** Keywords ***
Get Product Detail
    Create Session   toy_store   http://localhost:8000
    ${productDetail}=    Get Request    toy_store    /api/v1/product/2
    headers=&{accept}
    Request Should Be Successful    ${productDetail}
    Should Be Equal    ${productDetail.json()["id"]}    2
    Should Be Equal    ${productDetail.json()["product_name"]}    43 Piece
dinner Set
    Should Be Equal    ${productDetail.json()["product_price"]}    ${12.95}
```

Test Cases in Test Case Files

Order Dinner Set

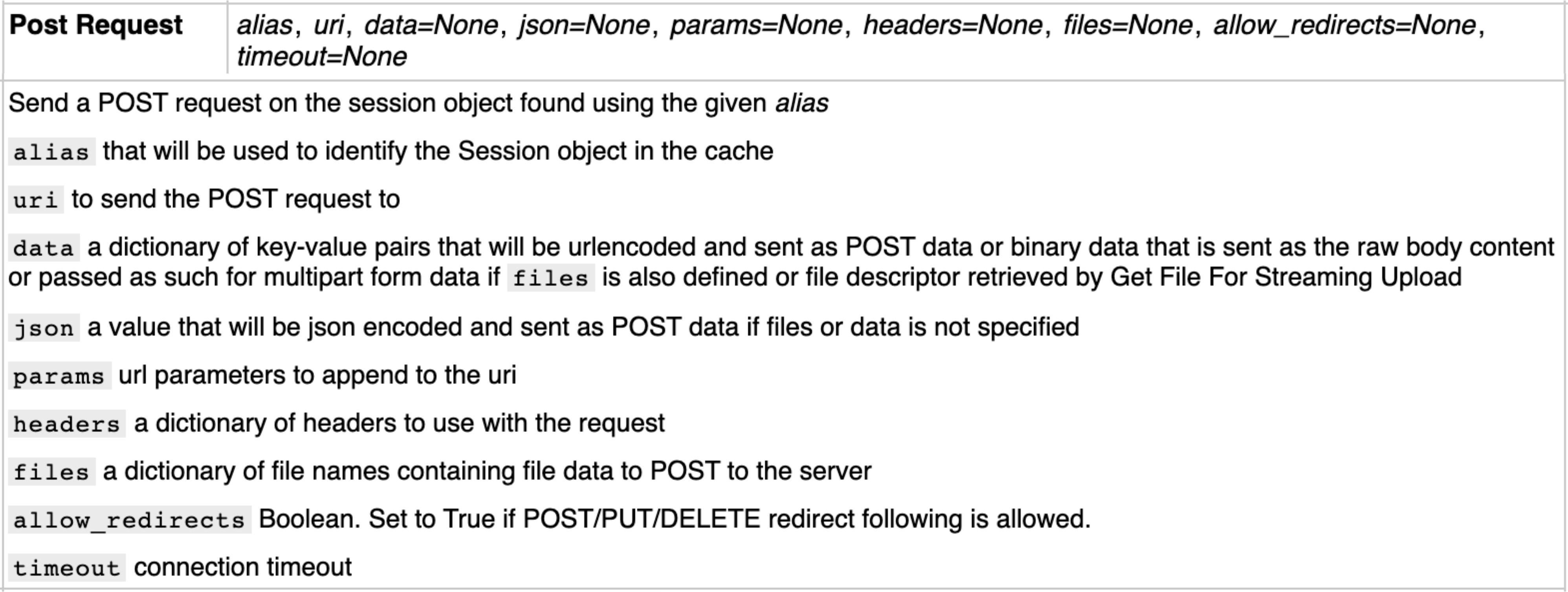
Assert Order Status



```
Request:
Order Dinner Set      list of [product id = 2, quantity = 1]
Order Dinner Set      shipping address
Order Dinner Set      headers = Accept=application/json
                                Content-Type=application/json

Response:
Assert                total price = 14.95
```

RequestsLibrary
Get Request API

lean more



Post Request	*alias*, *uri*, *data=None*, *json=None*, *params=None*, *headers=None*, *files=None*, *allow_redirects=None*, *timeout=None*

Send a POST request on the session object found using the given *alias*

`alias` that will be used to identify the Session object in the cache

`uri` to send the POST request to

`data` a dictionary of key-value pairs that will be urlencoded and sent as POST data or binary data that is sent as the raw body content or passed as such for multipart form data if `files` is also defined or file descriptor retrieved by Get File For Streaming Upload

`json` a value that will be json encoded and sent as POST data if files or data is not specified

`params` url parameters to append to the uri

`headers` a dictionary of headers to use with the request

`files` a dictionary of file names containing file data to POST to the server

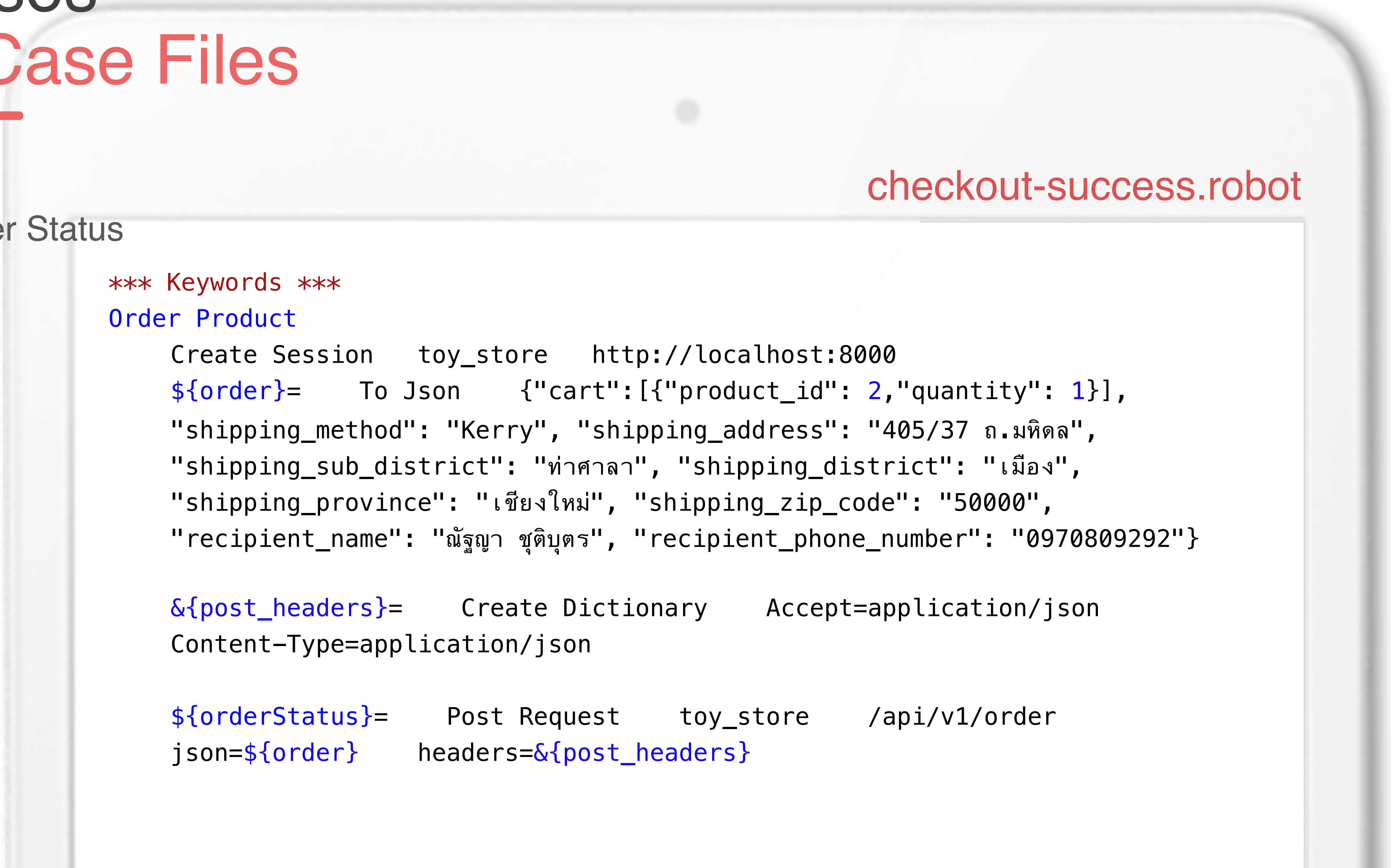
`allow_redirects` Boolean. Set to True if POST/PUT/DELETE redirect following is allowed.

`timeout` connection timeout

Test Cases
in Test Case Files

Order Product

Assert Order Status



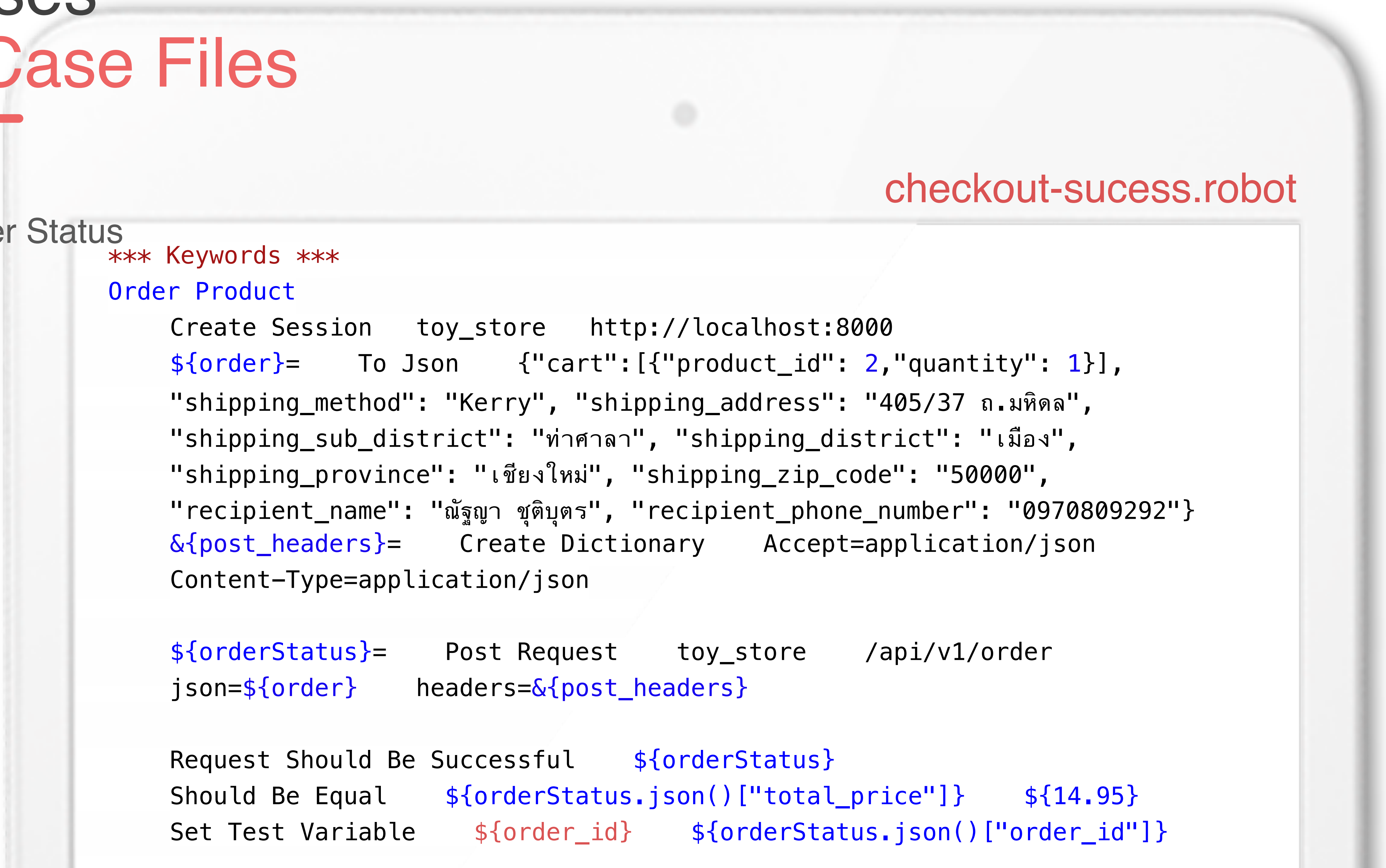
checkout-success.robot

```
*** Keywords ***
Order Product
    Create Session   toy_store   http://localhost:8000
    ${order}=    To Json    {"cart":[{"product_id": 2,"quantity": 1}],
    "shipping_method": "Kerry", "shipping_address": "405/37 ถ.มหิดล",
    "shipping_sub_district": "ท่าศาลา", "shipping_district": "เมือง",
    "shipping_province": "เชียงใหม่", "shipping_zip_code": "50000",
    "recipient_name": "ณัฐญา ชุติบุตร", "recipient_phone_number": "0970809292"}

    &{post_headers}=    Create Dictionary    Accept=application/json
    Content-Type=application/json

    ${orderStatus}=    Post Request    toy_store    /api/v1/order
    json=${order}    headers=&{post_headers}
```

Test Cases
in Test Case Files

Order Product

Assert Order Status



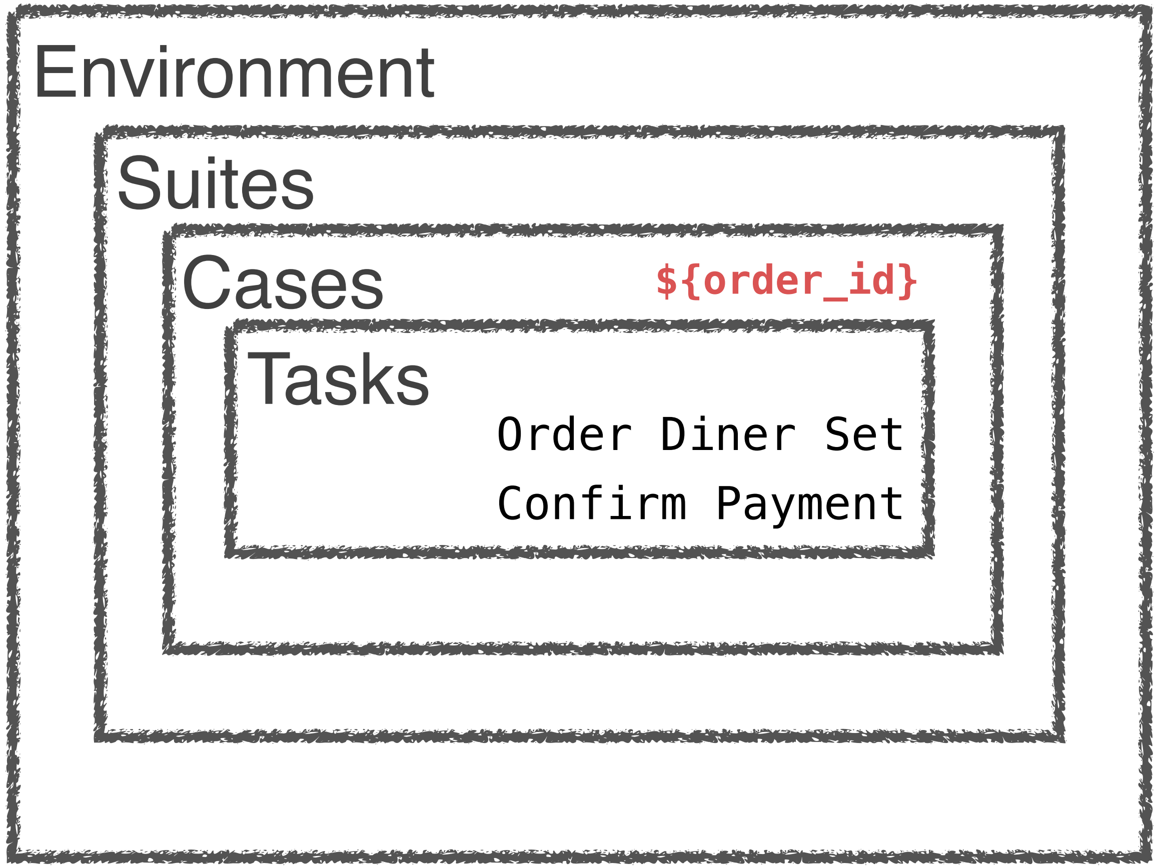
checkout-sucess.robot

```
*** Keywords ***
Order Product
    Create Session   toy_store   http://localhost:8000
    ${order}=    To Json    {"cart":[{"product_id": 2,"quantity": 1}],
    "shipping_method": "Kerry", "shipping_address": "405/37 ถ.มหิดล",
    "shipping_sub_district": "ท่าศาลา", "shipping_district": "เมือง",
    "shipping_province": "เชียงใหม่", "shipping_zip_code": "50000",
    "recipient_name": "ณัฐญา ชุติบุตร", "recipient_phone_number": "0970809292"}
    &{post_headers}=    Create Dictionary    Accept=application/json
    Content-Type=application/json

    ${orderStatus}=    Post Request    toy_store    /api/v1/order
    json=${order}    headers=&{post_headers}

    Request Should Be Successful    ${orderStatus}
    Should Be Equal    ${orderStatus.json()["total_price"]}    ${14.95}
    Set Test Variable    ${order_id}    ${orderStatus.json()["order_id"]}
```

Scope in Test Case Files

Robot Framework data is defined in different sections.

lean more



Environment

Suites

Cases

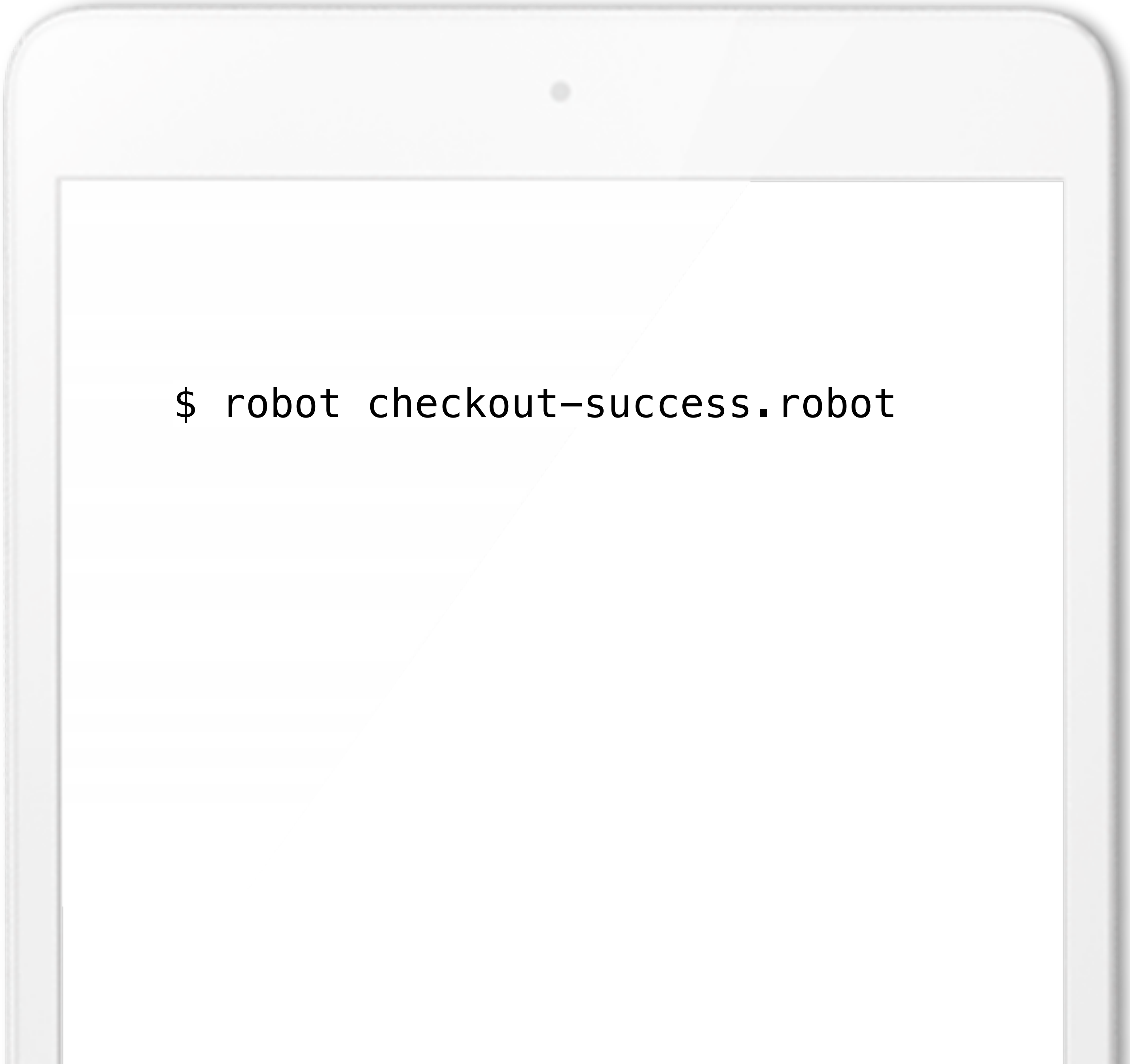
${order_id}

Tasks

Order Diner Set

Confirm Payment

Run
Test Case Files

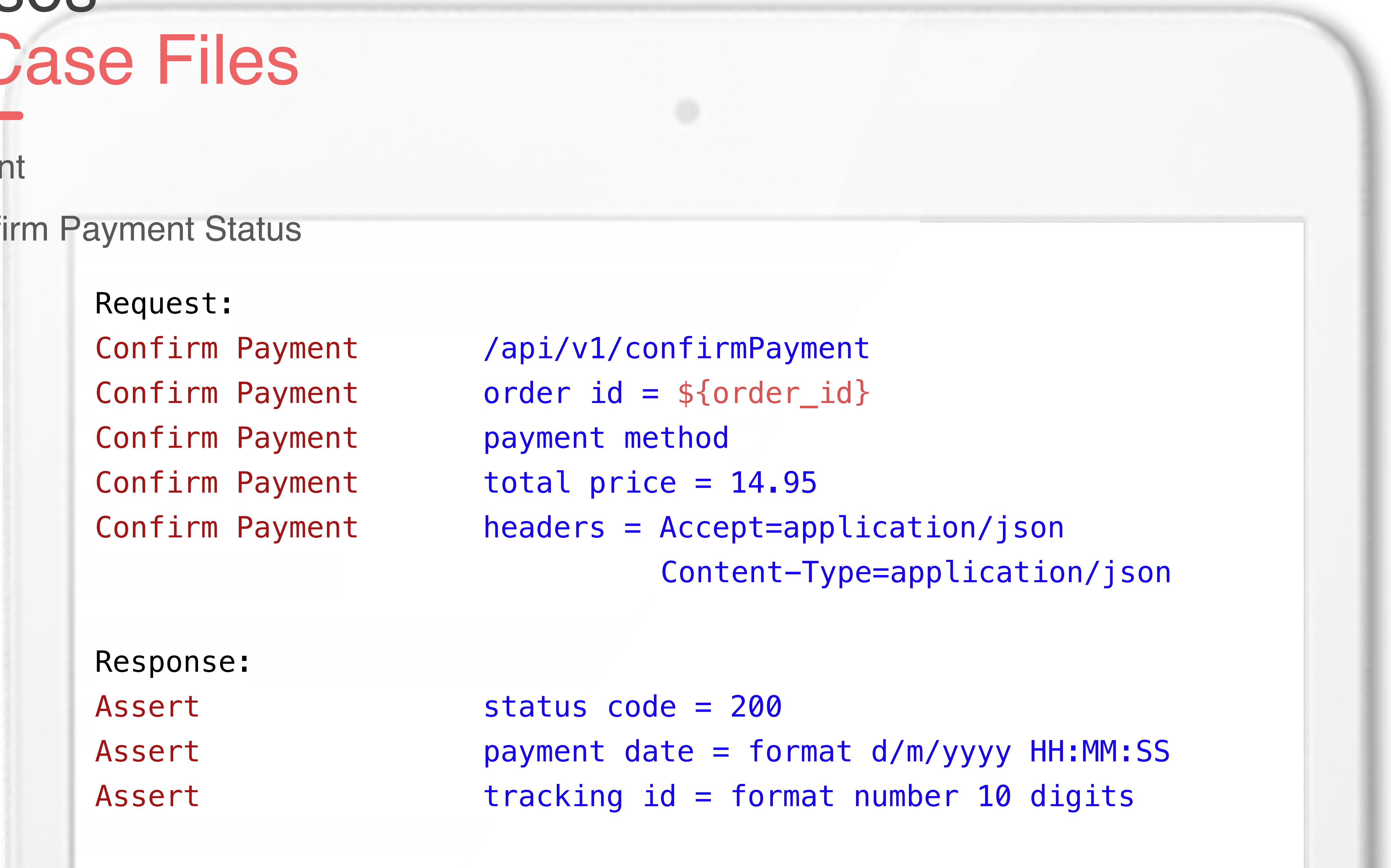


```
$ robot checkout-success.robot
```

Test Cases in Test Case Files

Confirm Payment

Assert Confirm Payment Status

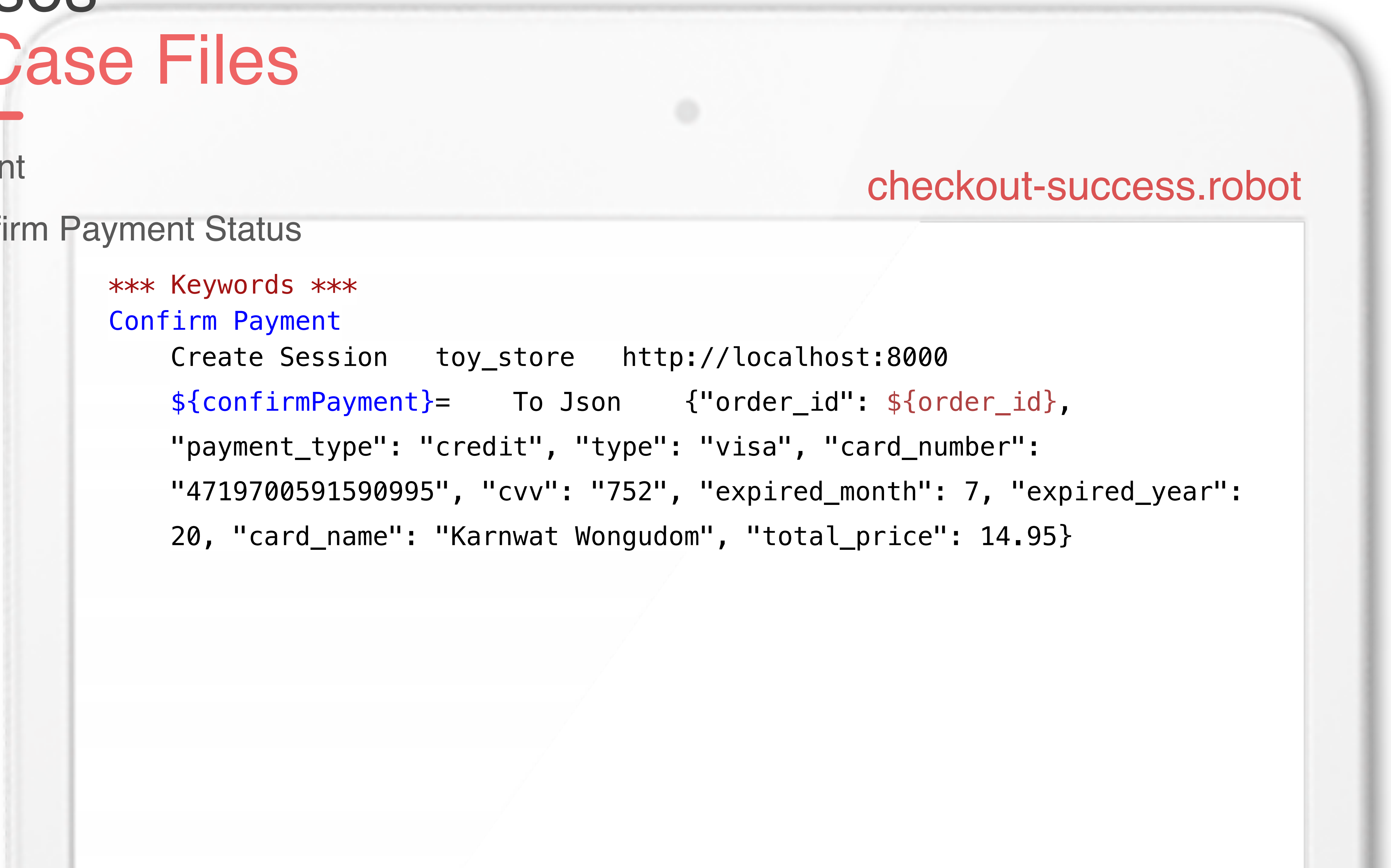


```
Request:
Confirm Payment       /api/v1/confirmPayment
Confirm Payment       order id = ${order_id}
Confirm Payment       payment method
Confirm Payment       total price = 14.95
Confirm Payment       headers = Accept=application/json
                                Content-Type=application/json


Response:
Assert                status code = 200
Assert                payment date = format d/m/yyyy HH:MM:SS
Assert                tracking id = format number 10 digits
```

Test Cases
in Test Case Files

Confirm Payment

Assert Confirm Payment Status



checkout-success.robot

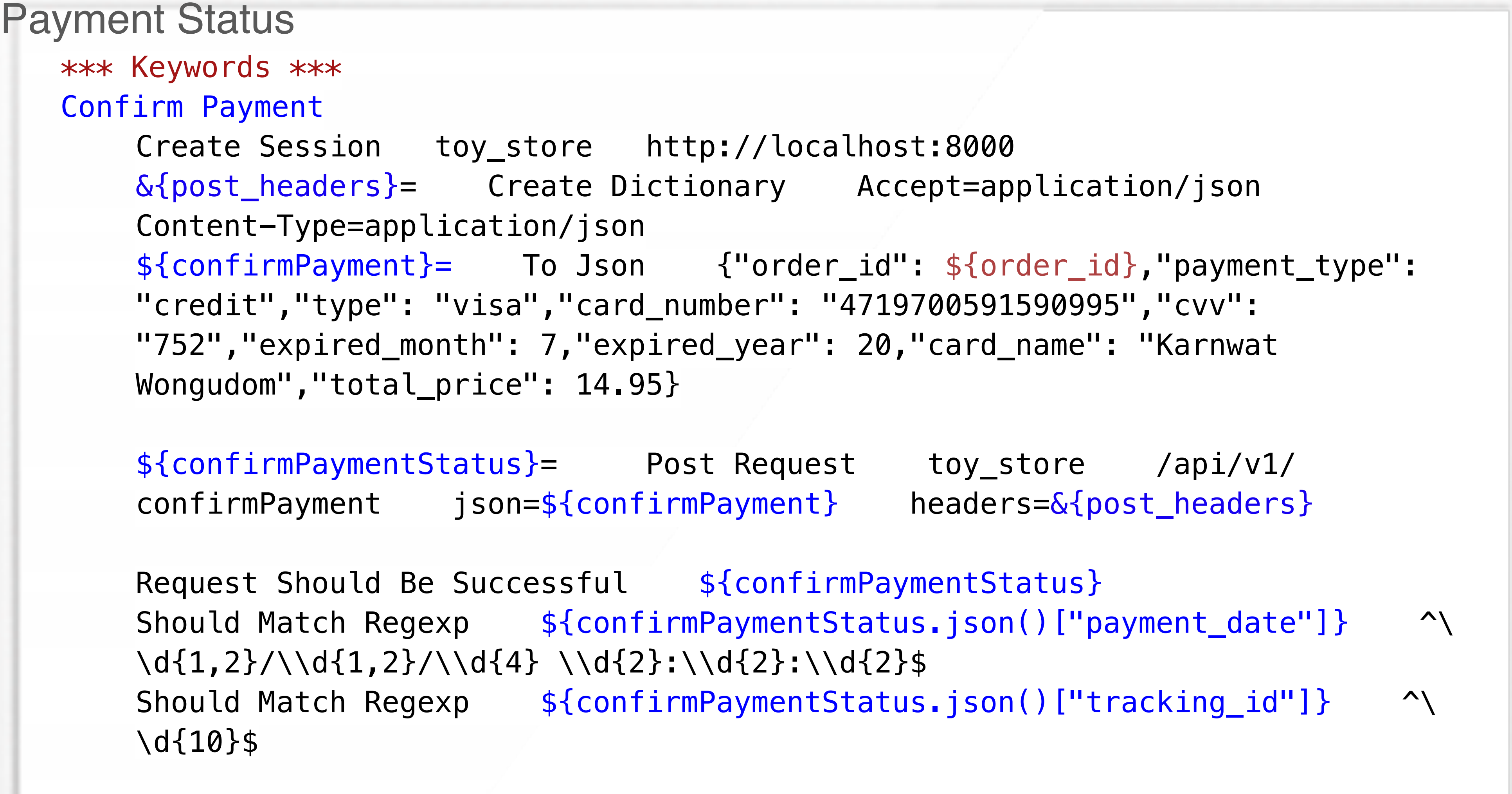
```
*** Keywords ***
Confirm Payment
    Create Session    toy_store    http://localhost:8000
    ${confirmPayment}=    To Json    {"order_id": ${order_id},
    "payment_type": "credit", "type": "visa", "card_number":
    "4719700591590995", "cvv": "752", "expired_month": 7, "expired_year":
    20, "card_name": "Karnwat Wongudom", "total_price": 14.95}
```

Test Cases
in Test Case Files

Confirm Payment

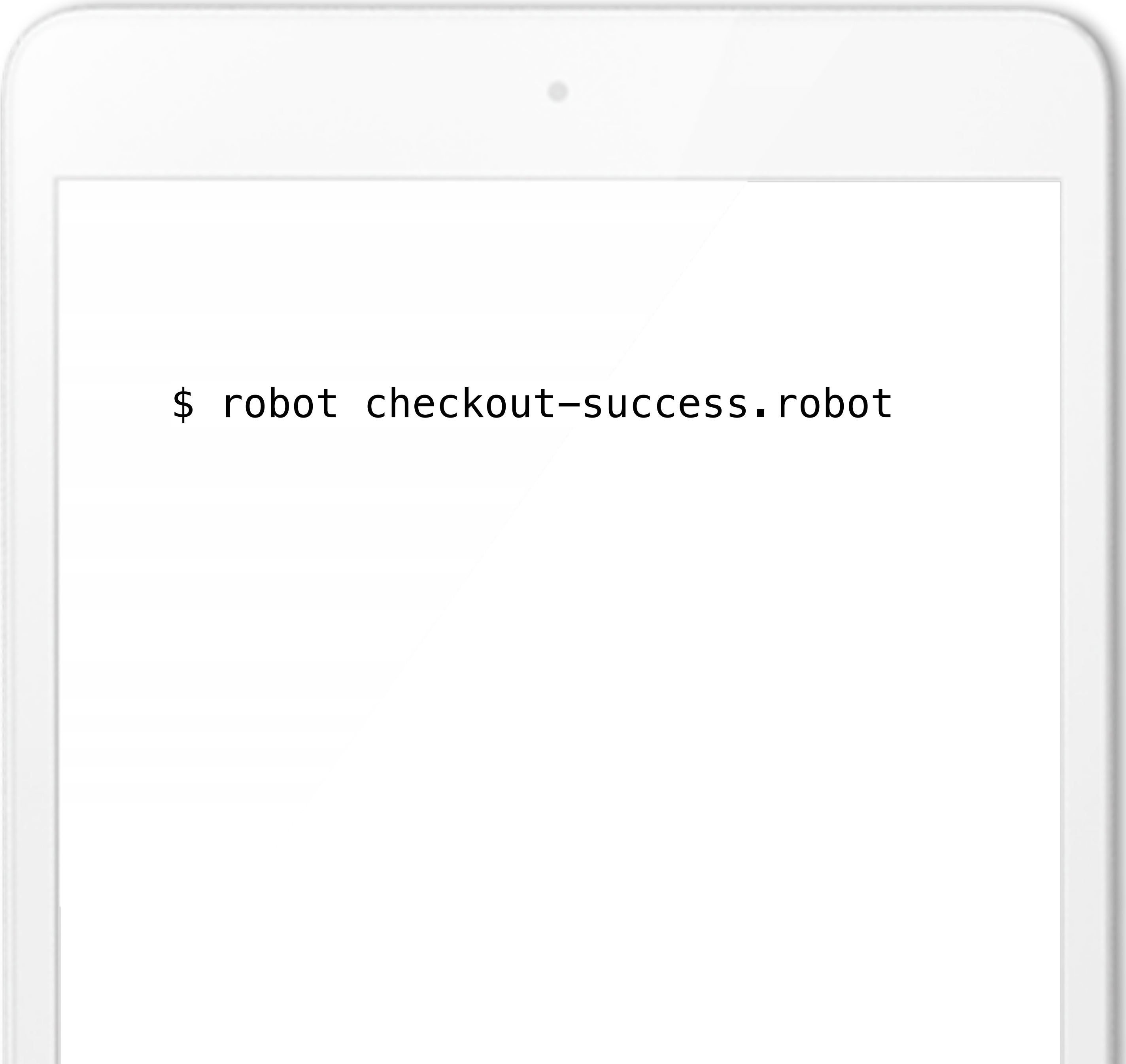
Assert Confirm Payment Status

checkout-success.robot



```
*** Keywords ***
Confirm Payment
    Create Session   toy_store   http://localhost:8000
    &{post_headers}=    Create Dictionary    Accept=application/json
    Content-Type=application/json
    ${confirmPayment}=    To Json    {"order_id": ${order_id},"payment_type":
    "credit","type": "visa","card_number": "4719700591590995","cvv":
    "752","expired_month": 7,"expired_year": 20,"card_name": "Karnwat
    Wongudom","total_price": 14.95}

    ${confirmPaymentStatus}=     Post Request    toy_store    /api/v1/
    confirmPayment    json=${confirmPayment}    headers=&{post_headers}

    Request Should Be Successful    ${confirmPaymentStatus}
    Should Match Regexp    ${confirmPaymentStatus.json()["payment_date"]}    ^\
    \d{1,2}/\\d{1,2}/\\d{4} \\d{2}:\\d{2}:\\d{2}$
    Should Match Regexp    ${confirmPaymentStatus.json()["tracking_id"]}    ^\
    \d{10}$
```

Run
Test Case Files



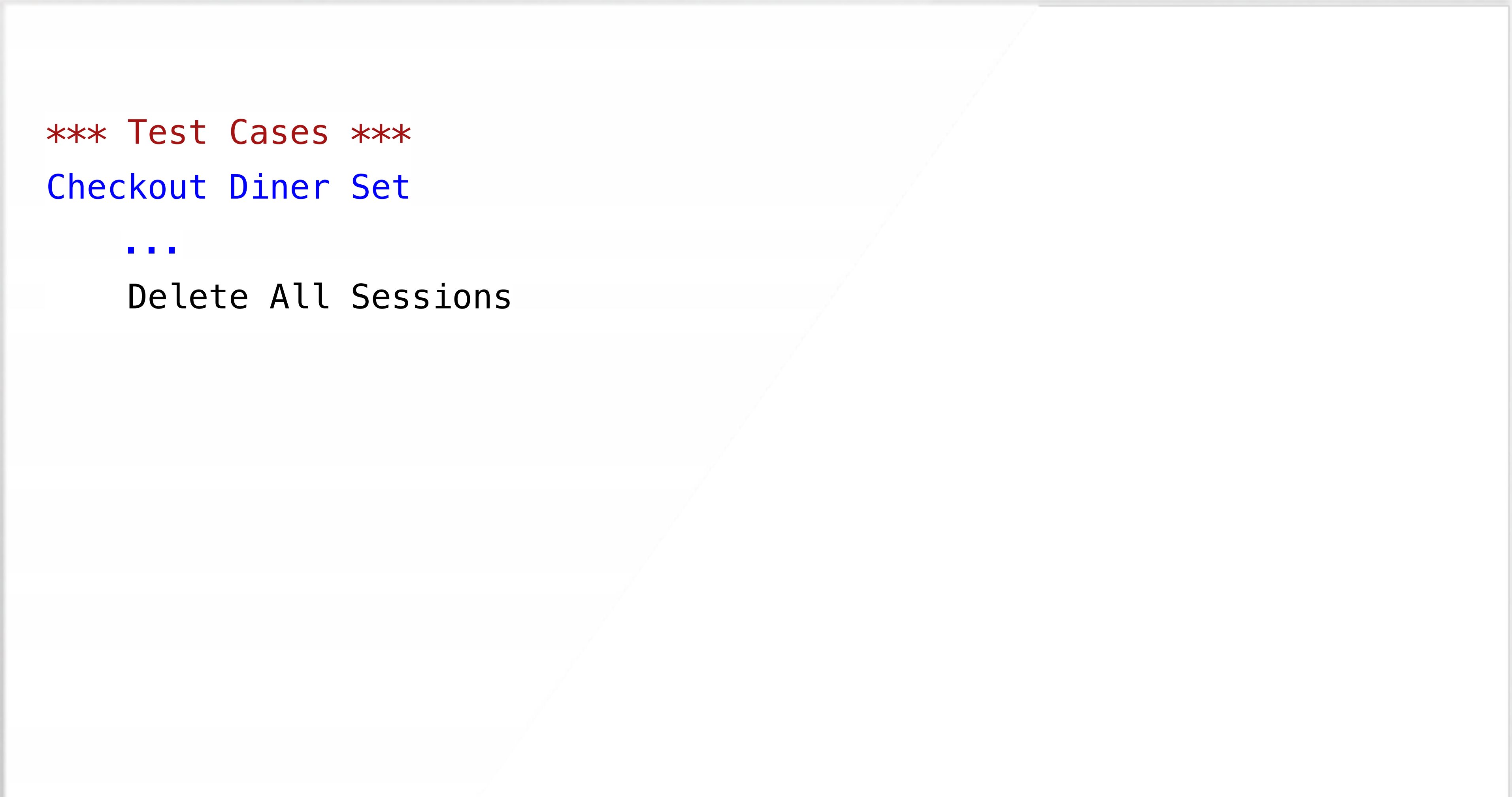
```
$ robot checkout-success.robot
```

Test Cases in Test Case Files

Create Checkout Diner Set(Test case)

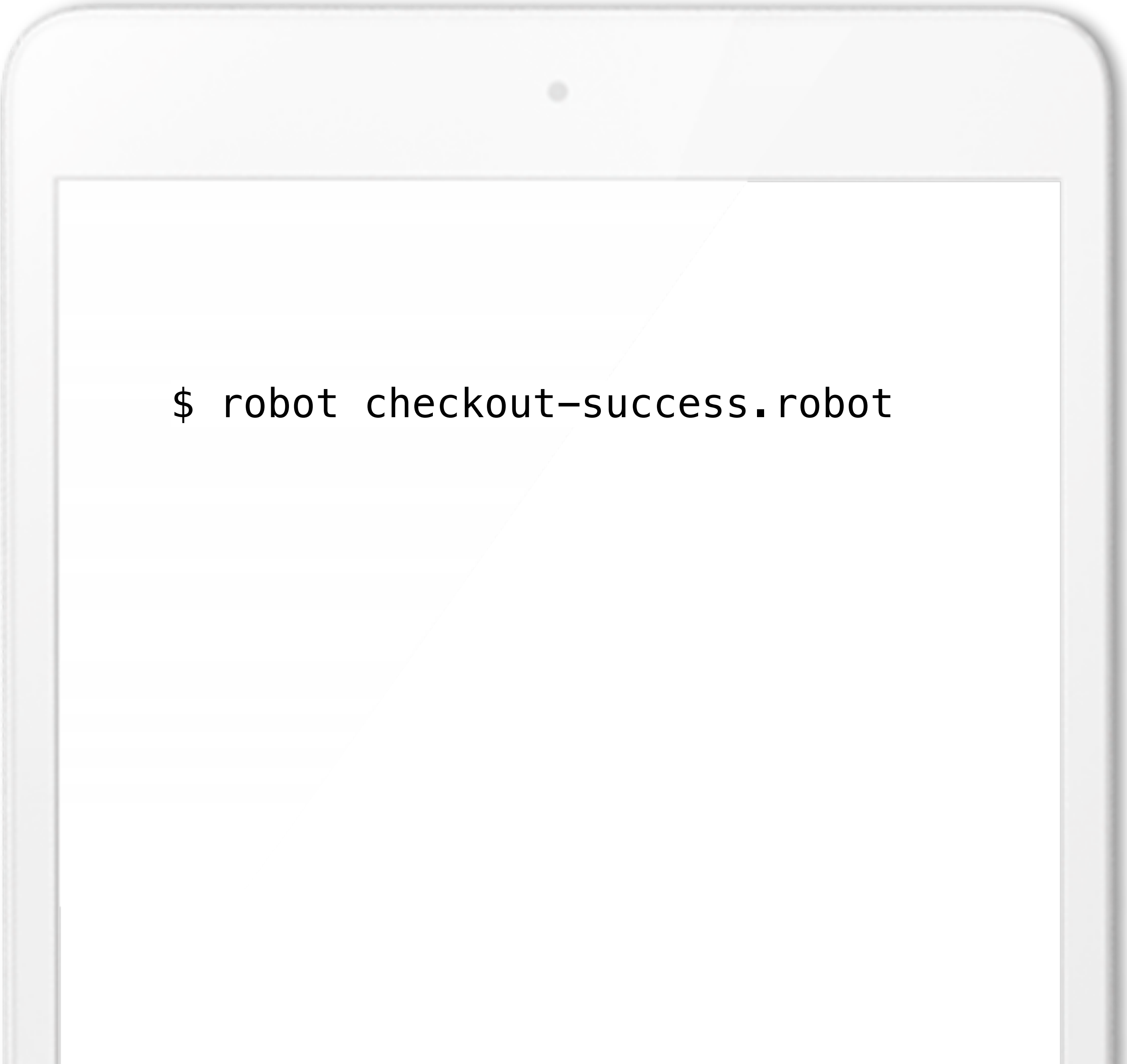
Delete Session

checkout-success.robot



```
*** Test Cases ***
Checkout Diner Set
    ...
    Delete All Sessions
```

Run
Test Case Files



```
$ robot checkout-success.robot
```

HTTP RequestsLibrary(Python)

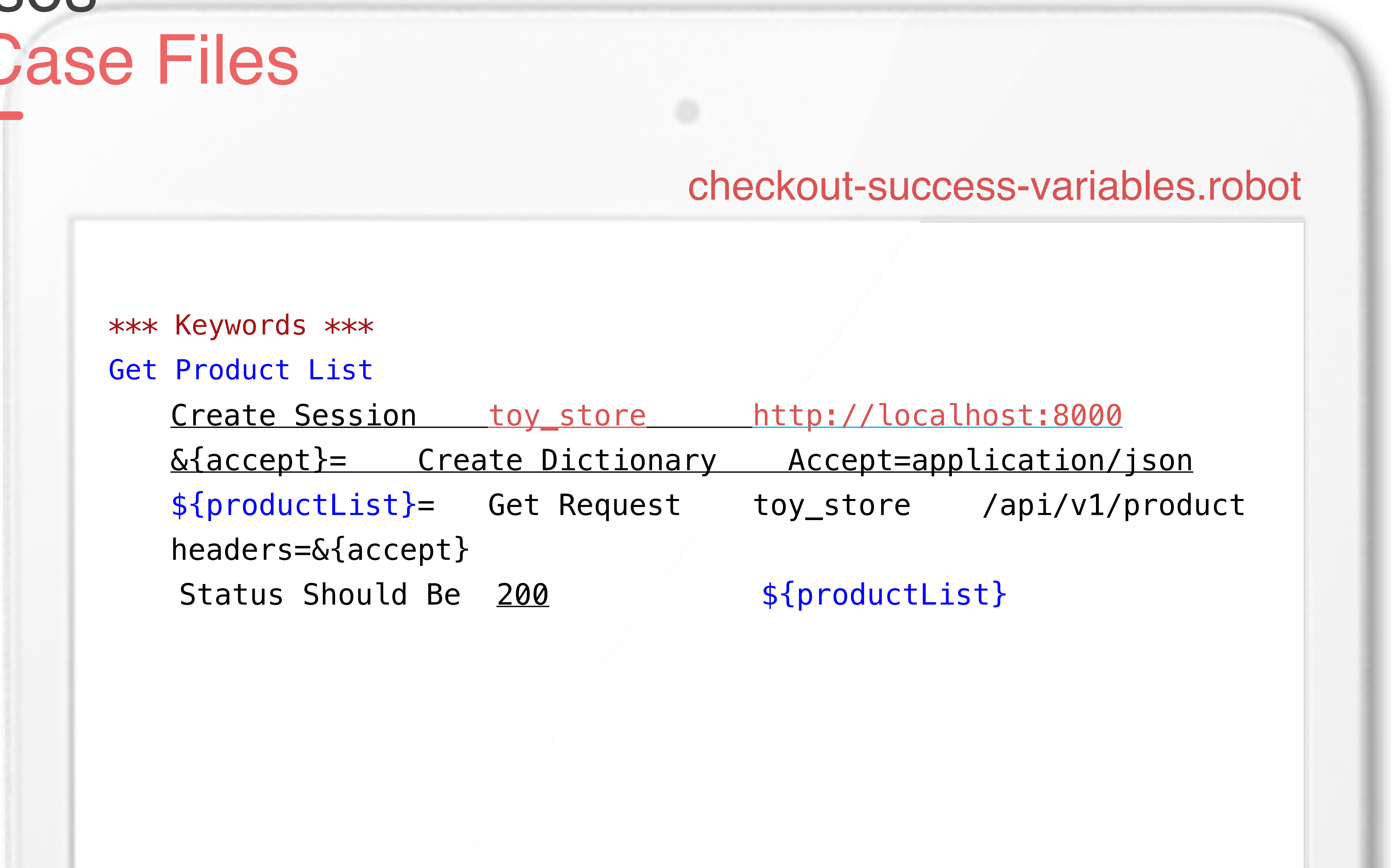
*** Variables ***

lean more

© 2020 Siam Chamnankit Company Limited

Test Cases
in Test Case Files

Hard Code!

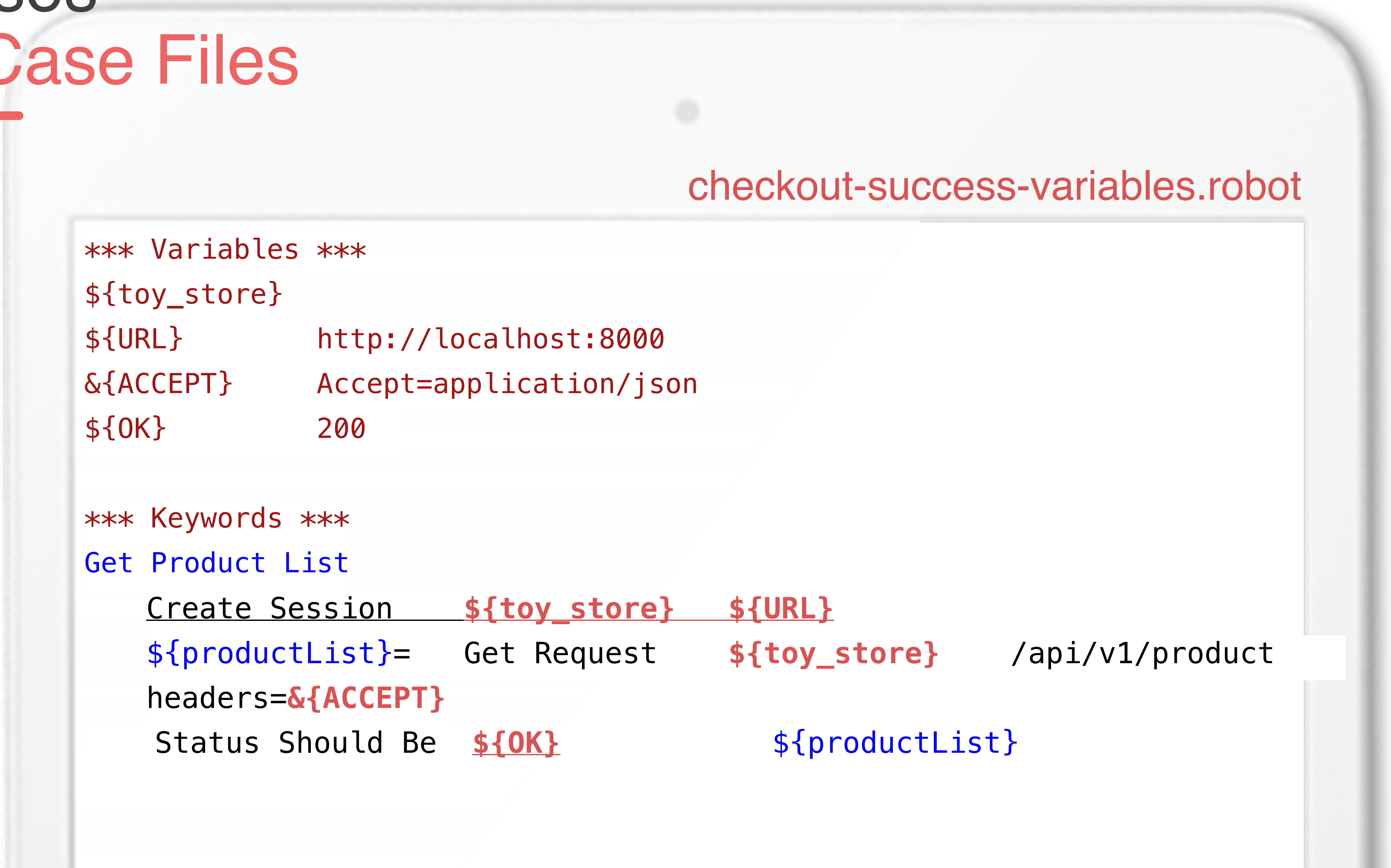


checkout-success-variables.robot

```
*** Keywords ***
Get Product List
    Create Session    toy_store    http://localhost:8000
    &{accept}=    Create Dictionary    Accept=application/json
    ${productList}=   Get Request    toy_store    /api/v1/product
    headers=&{accept}
     Status Should Be  200              ${productList}
```

Test Cases in Test Case Files

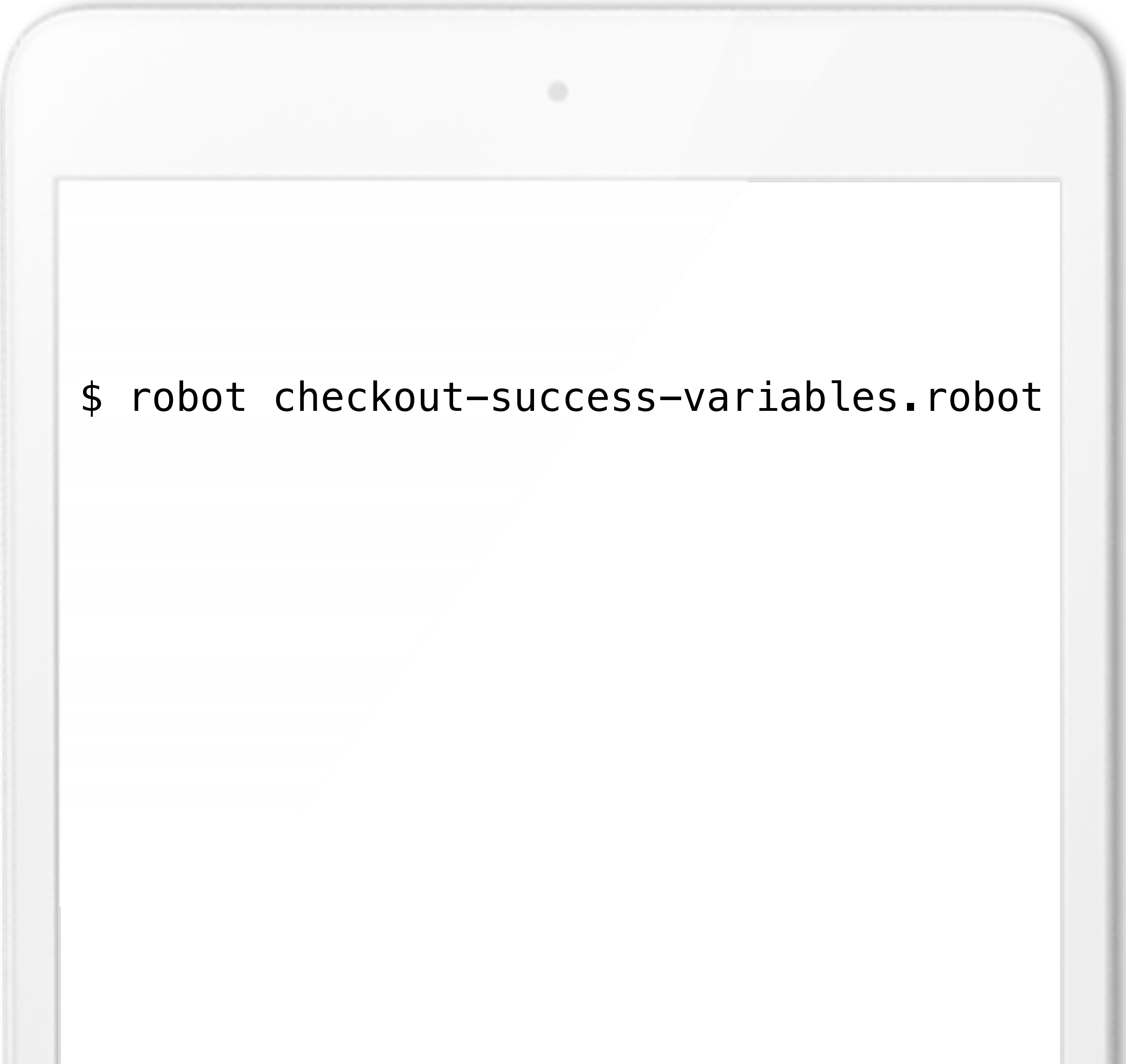
Hard Code!



checkout-success-variables.robot

```
*** Variables ***
${toy_store}
${URL}          http://localhost:8000
&{ACCEPT}       Accept=application/json
${OK}           200

*** Keywords ***
Get Product List
    Create Session     ${toy_store}    ${URL}
    ${productList}=    Get Request    ${toy_store}    /api/v1/product
    headers=&{ACCEPT}
     Status Should Be  ${OK}             ${productList}
```

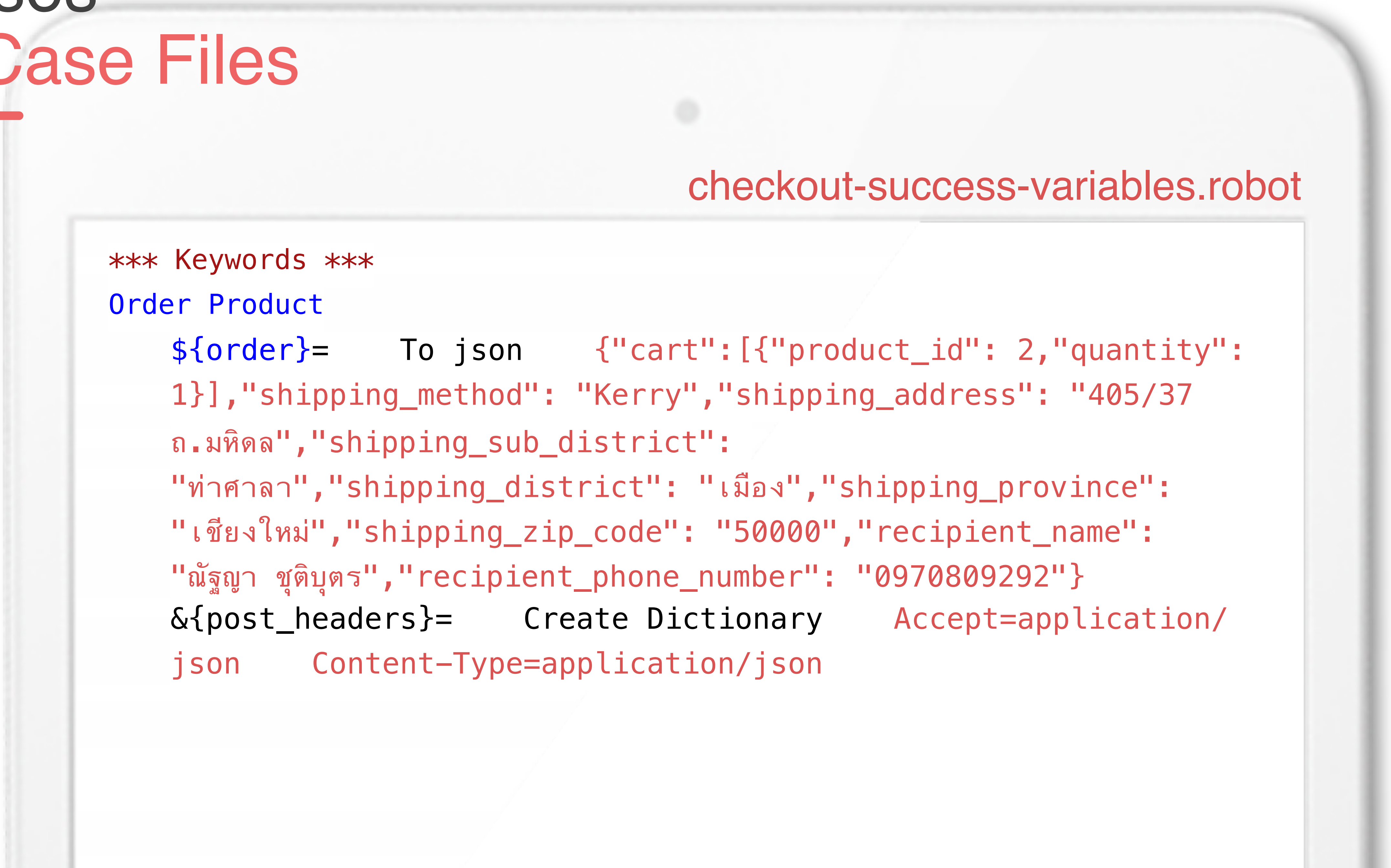
Run
Test Case Files



```
$ robot checkout-success-variables.robot
```

Test Cases in Test Case Files

Hard Code!

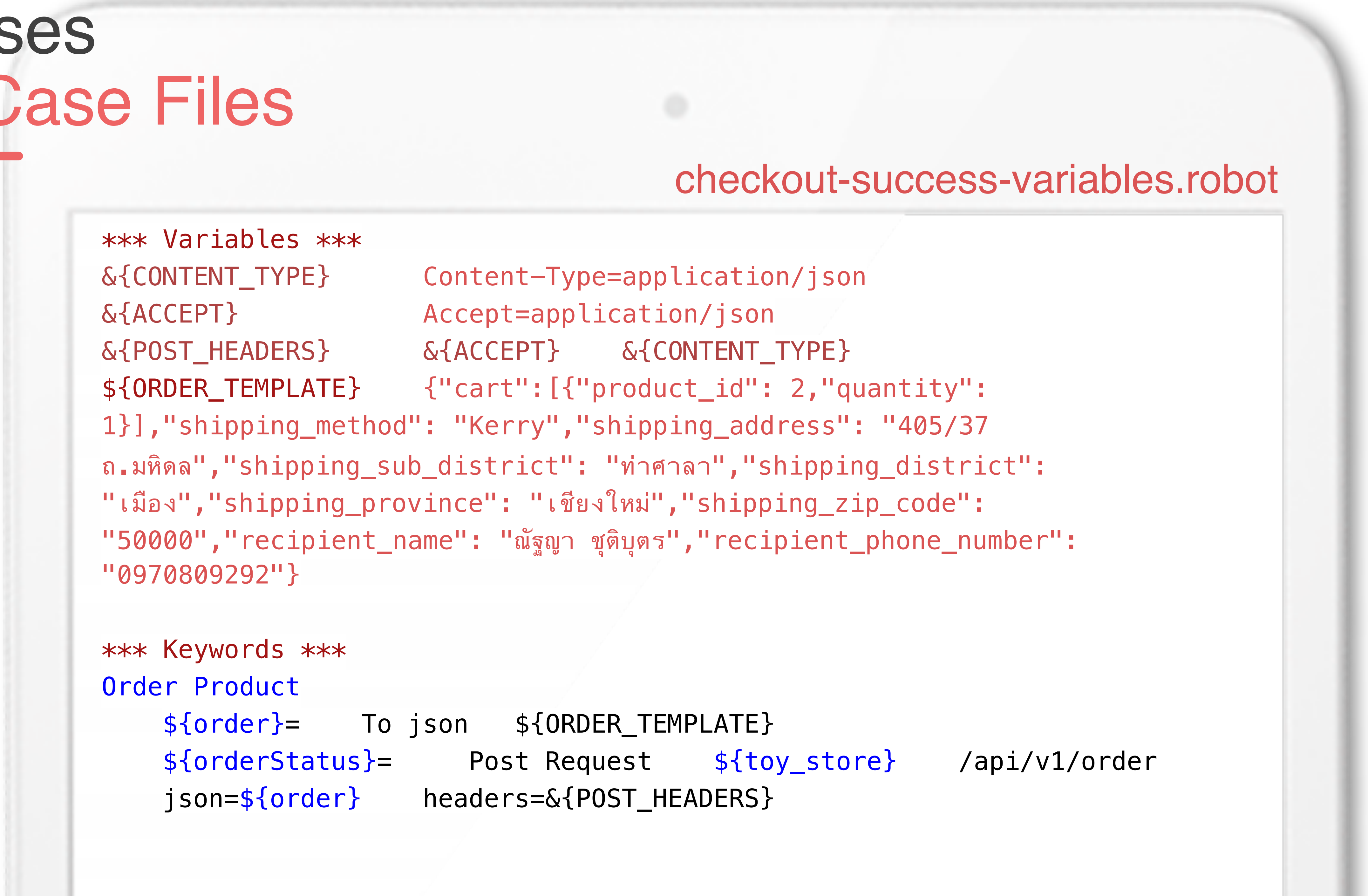


checkout-success-variables.robot

```
*** Keywords ***
Order Product
    ${order}=    To json    {"cart":[{"product_id": 2,"quantity":
    1}],"shipping_method": "Kerry","shipping_address": "405/37
    ถ.มหิดล","shipping_sub_district":
    "ท่าศาลา","shipping_district": "เมือง","shipping_province":
    "เชียงใหม่","shipping_zip_code": "50000","recipient_name":
    "ณัฐญา ชุติบุตร","recipient_phone_number": "0970809292"}
    &{post_headers}=    Create Dictionary    Accept=application/
    json    Content-Type=application/json
```

Test Cases
in Test Case Files

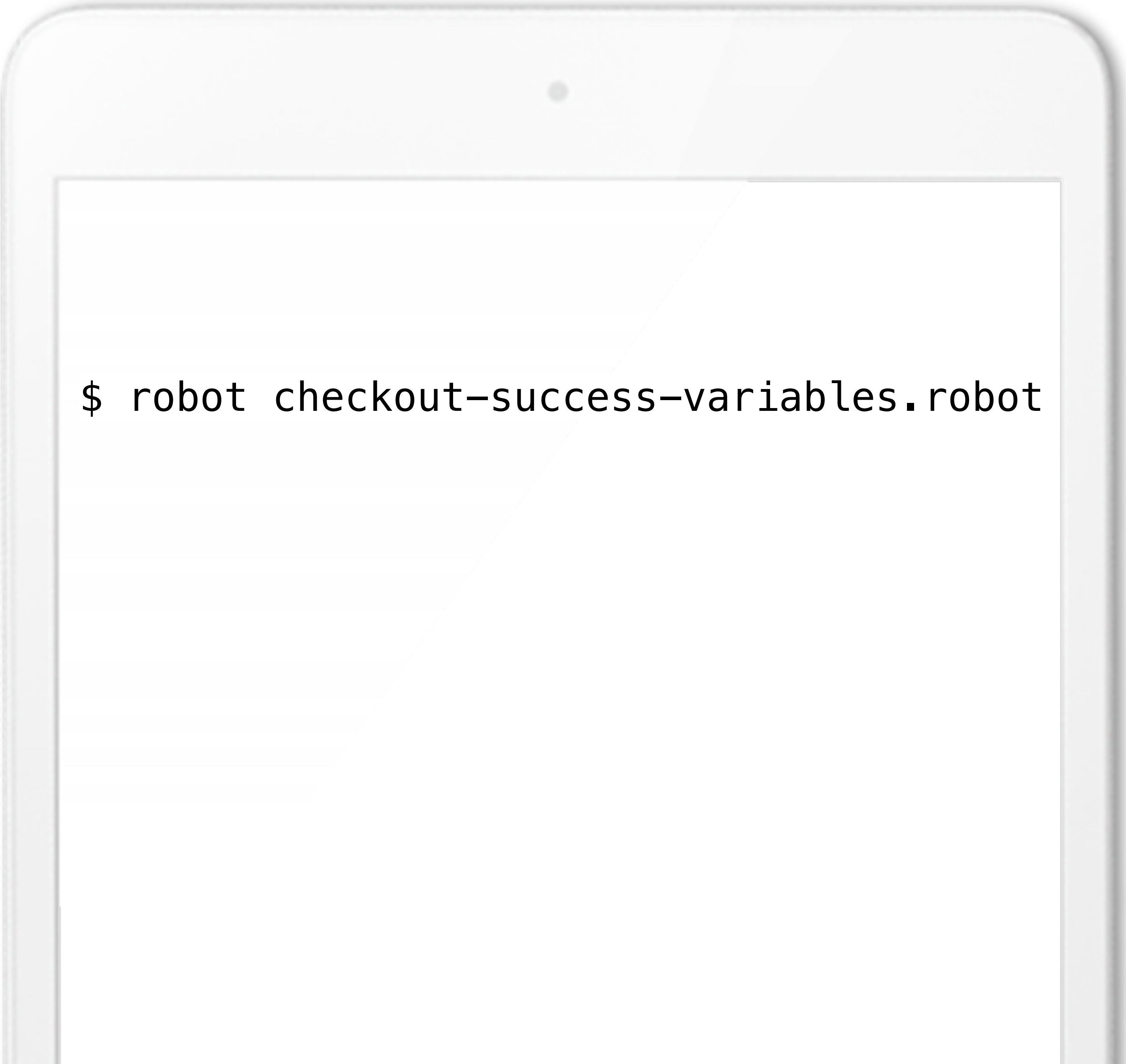
Hard Code!



checkout-success-variables.robot

```
*** Variables ***
&{CONTENT_TYPE}      Content-Type=application/json
&{ACCEPT}            Accept=application/json
&{POST_HEADERS}      &{ACCEPT}    &{CONTENT_TYPE}
${ORDER_TEMPLATE}    {"cart":[{"product_id": 2,"quantity": 
1}],"shipping_method": "Kerry","shipping_address": "405/37 
ถ.มหิดล","shipping_sub_district": "ท่าศาลา","shipping_district": 
"เมือง","shipping_province": "เชียงใหม่","shipping_zip_code": 
"50000","recipient_name": "ณัฐญา ชุติบุตร","recipient_phone_number": 
"0970809292"}

*** Keywords ***
Order Product
    ${order}=    To json    ${ORDER_TEMPLATE}
    ${orderStatus}=    Post Request    ${toy_store}    /api/v1/order
    json=${order}    headers=&{POST_HEADERS}
```

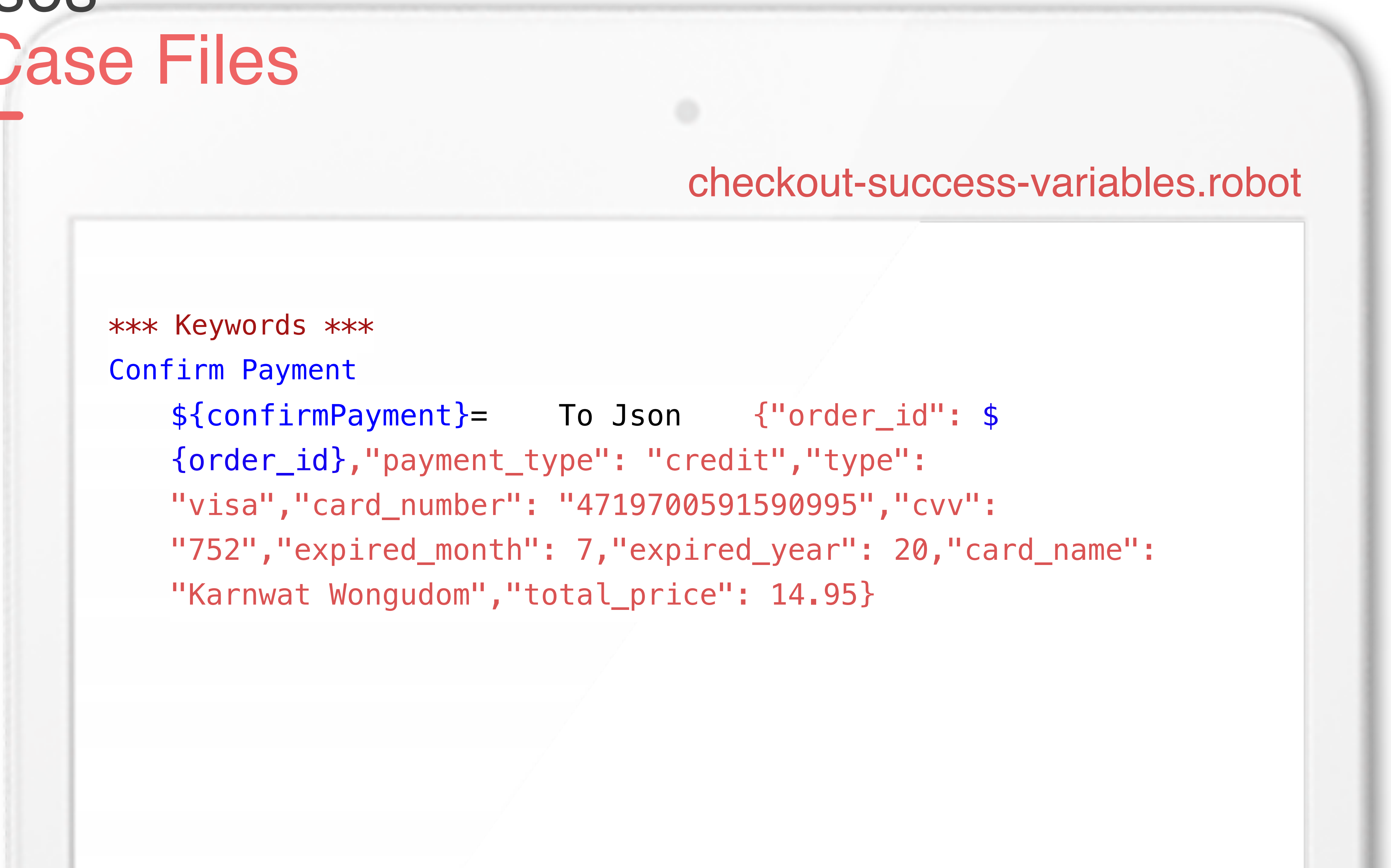
Run
Test Case Files



```
$ robot checkout-success-variables.robot
```

Test Cases
in Test Case Files

Hard Code!

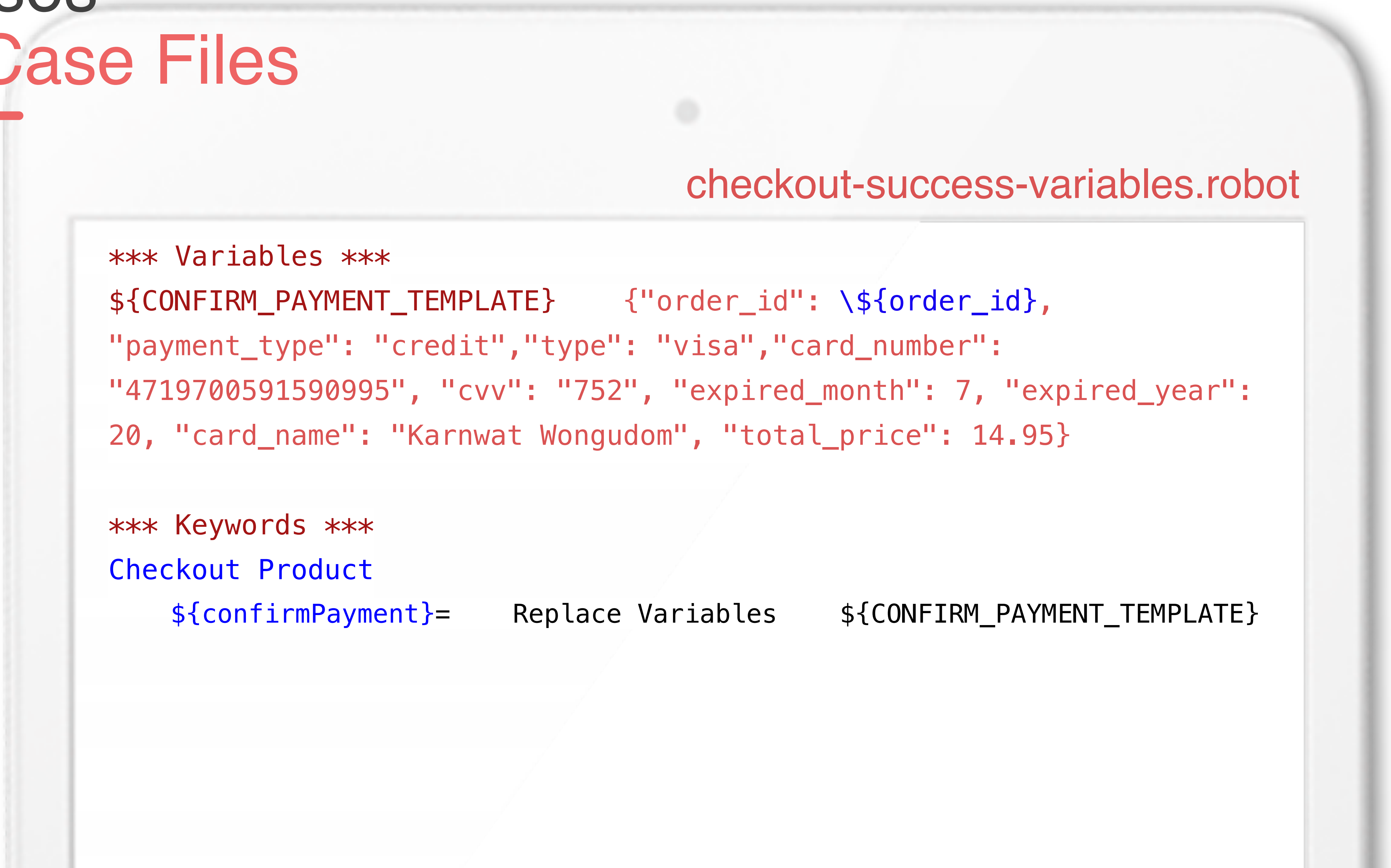


checkout-success-variables.robot

```
*** Keywords ***
Confirm Payment
    ${confirmPayment}=    To Json    {"order_id": $
    {order_id},"payment_type": "credit","type":
    "visa","card_number": "4719700591590995","cvv":
    "752","expired_month": 7,"expired_year": 20,"card_name":
    "Karnwat Wongudom","total_price": 14.95}
```

Test Cases
in Test Case Files

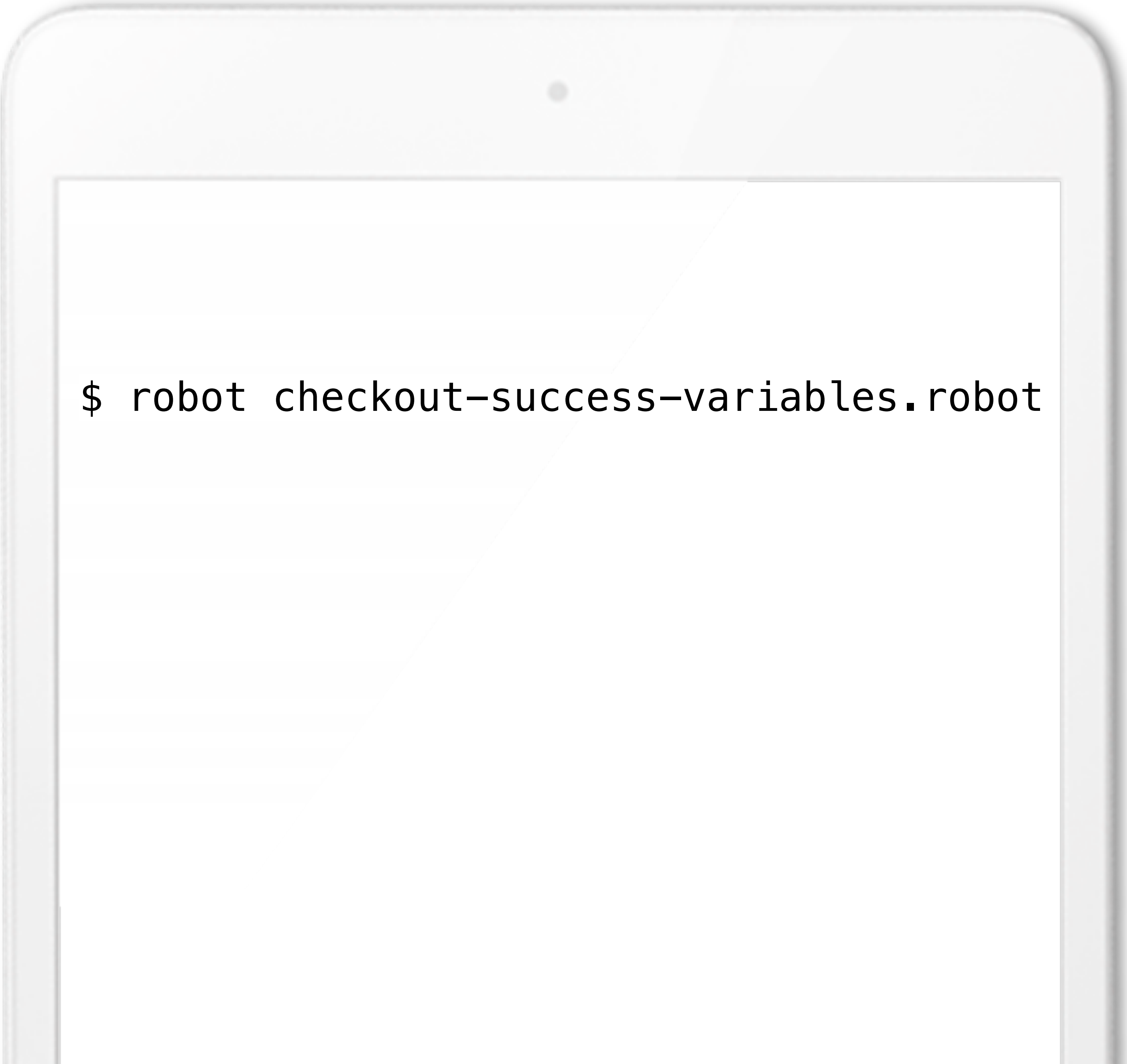
Hard Code!



checkout-success-variables.robot

```
*** Variables ***
${CONFIRM_PAYMENT_TEMPLATE}    {"order_id": \${order_id},
"payment_type": "credit","type": "visa","card_number":
"4719700591590995", "cvv": "752", "expired_month": 7, "expired_year":
20, "card_name": "Karnwat Wongudom", "total_price": 14.95}

*** Keywords ***
Checkout Product
    ${confirmPayment}=    Replace Variables    ${CONFIRM_PAYMENT_TEMPLATE}
```

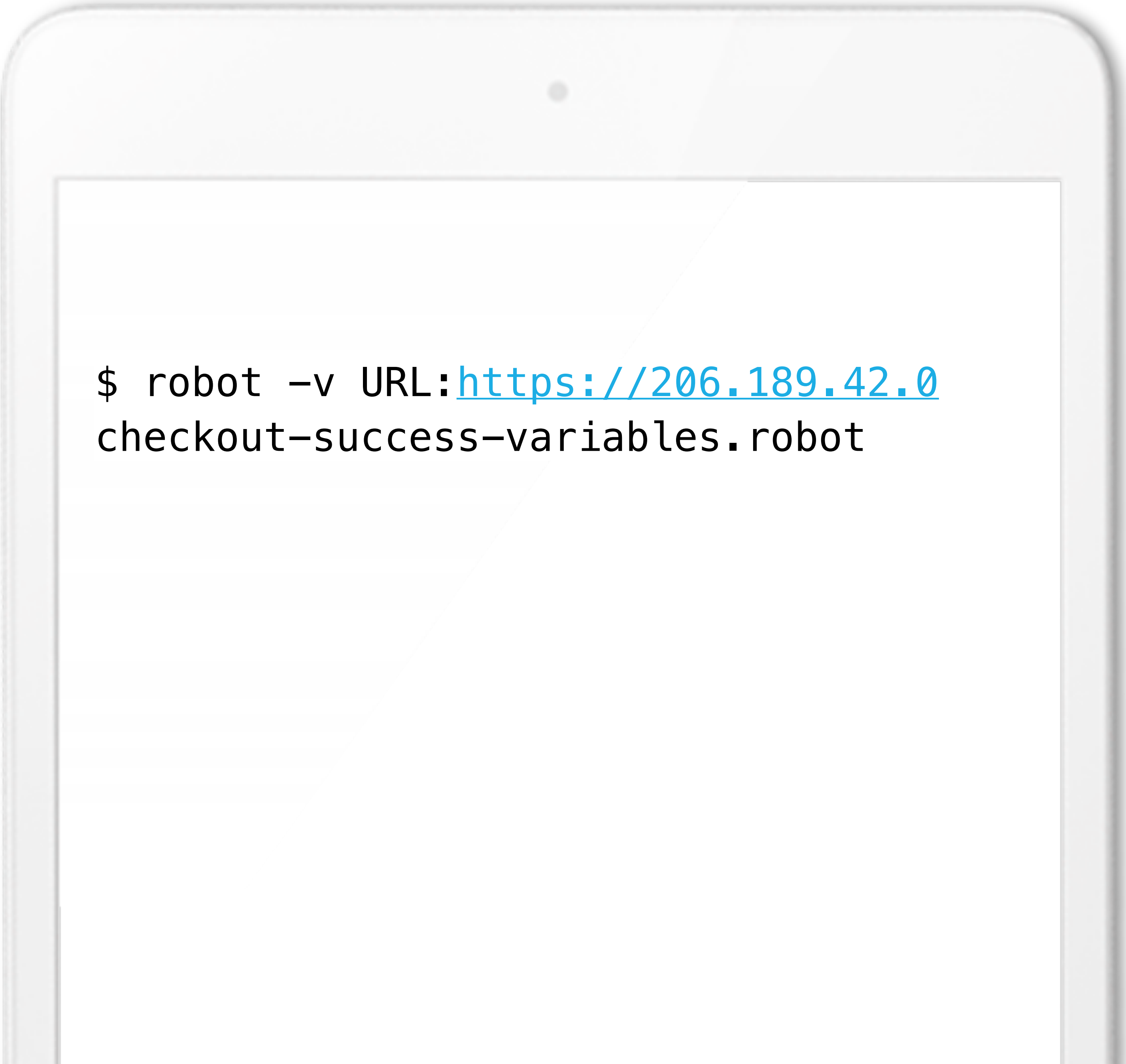
Run
Test Case Files



```
$ robot checkout-success-variables.robot
```

Run
with Options

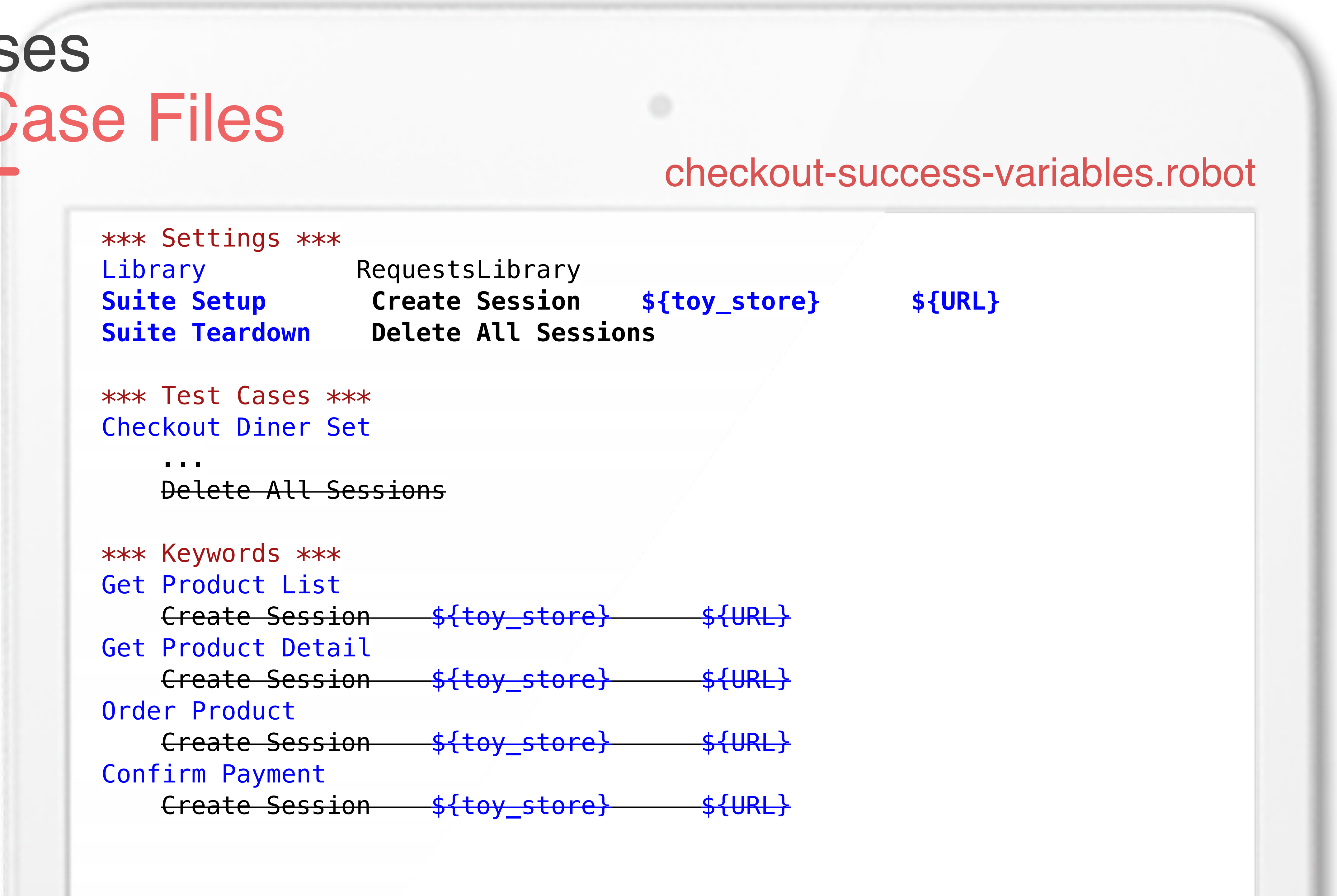
lean more



```
$ robot -v URL:https://206.189.42.0
checkout-success-variables.robot
```

Test Cases in Test Case Files

Hard Code!



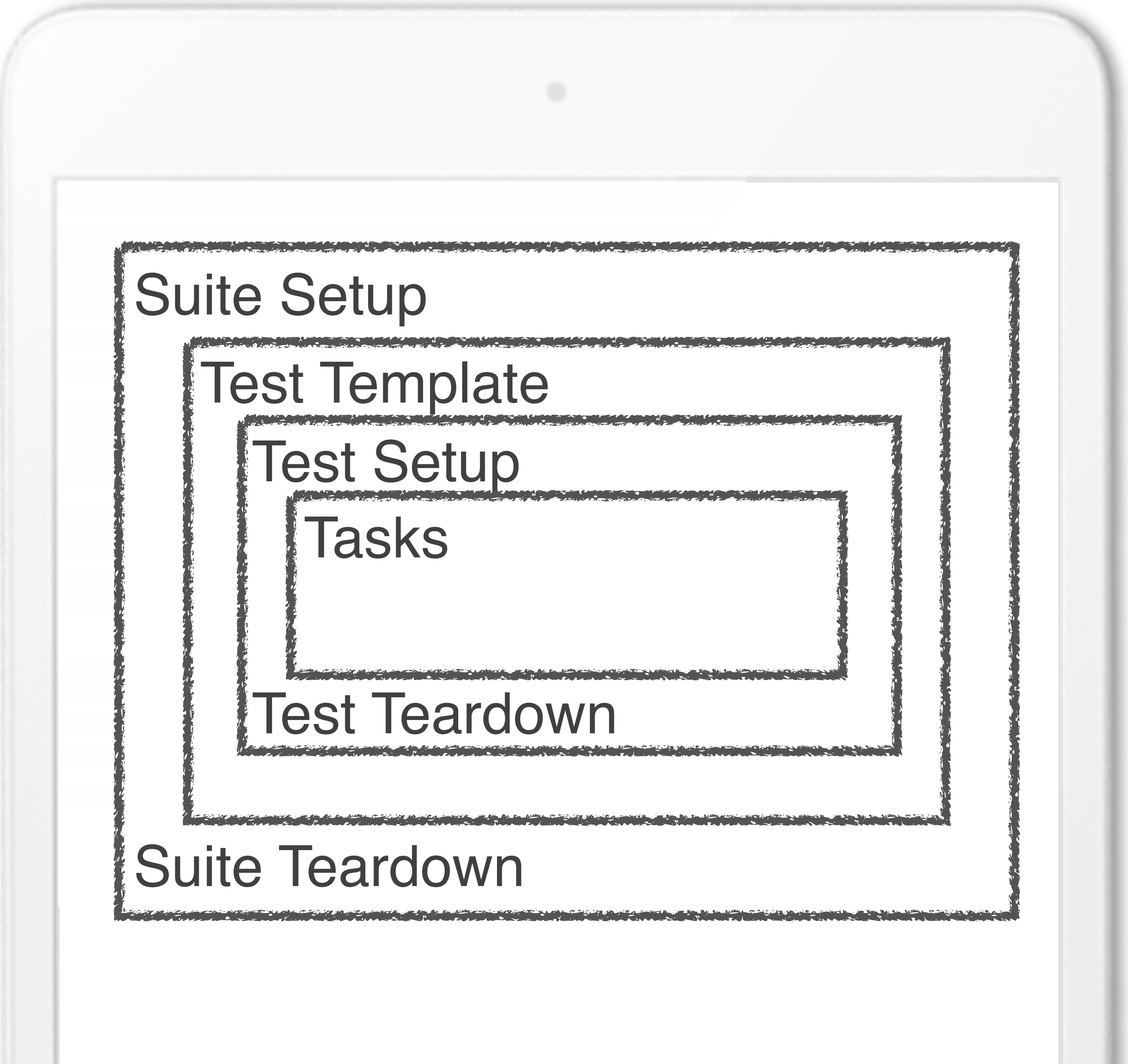
checkout-success-variables.robot

```
*** Settings ***
Library                 RequestsLibrary
Suite Setup             Create Session    ${toy_store}        ${URL}
Suite Teardown          Delete All Sessions

*** Test Cases ***
Checkout Diner Set
    ...
    ~~Delete All Sessions~~

*** Keywords ***
Get Product List
    ~~Create Session    ${toy_store}        ${URL}~~
Get Product Detail
    ~~Create Session    ${toy_store}        ${URL}~~
Order Product
    ~~Create Session    ${toy_store}        ${URL}~~
Confirm Payment
    ~~Create Session    ${toy_store}        ${URL}~~
```

Execution Flow in Test Case Files

Robot Framework data is defined in different sections.

lean more



Suite Setup

Test Template

Test Setup

Tasks

Test Teardown

Suite Teardown

HTTP RequestsLibrary(Python)

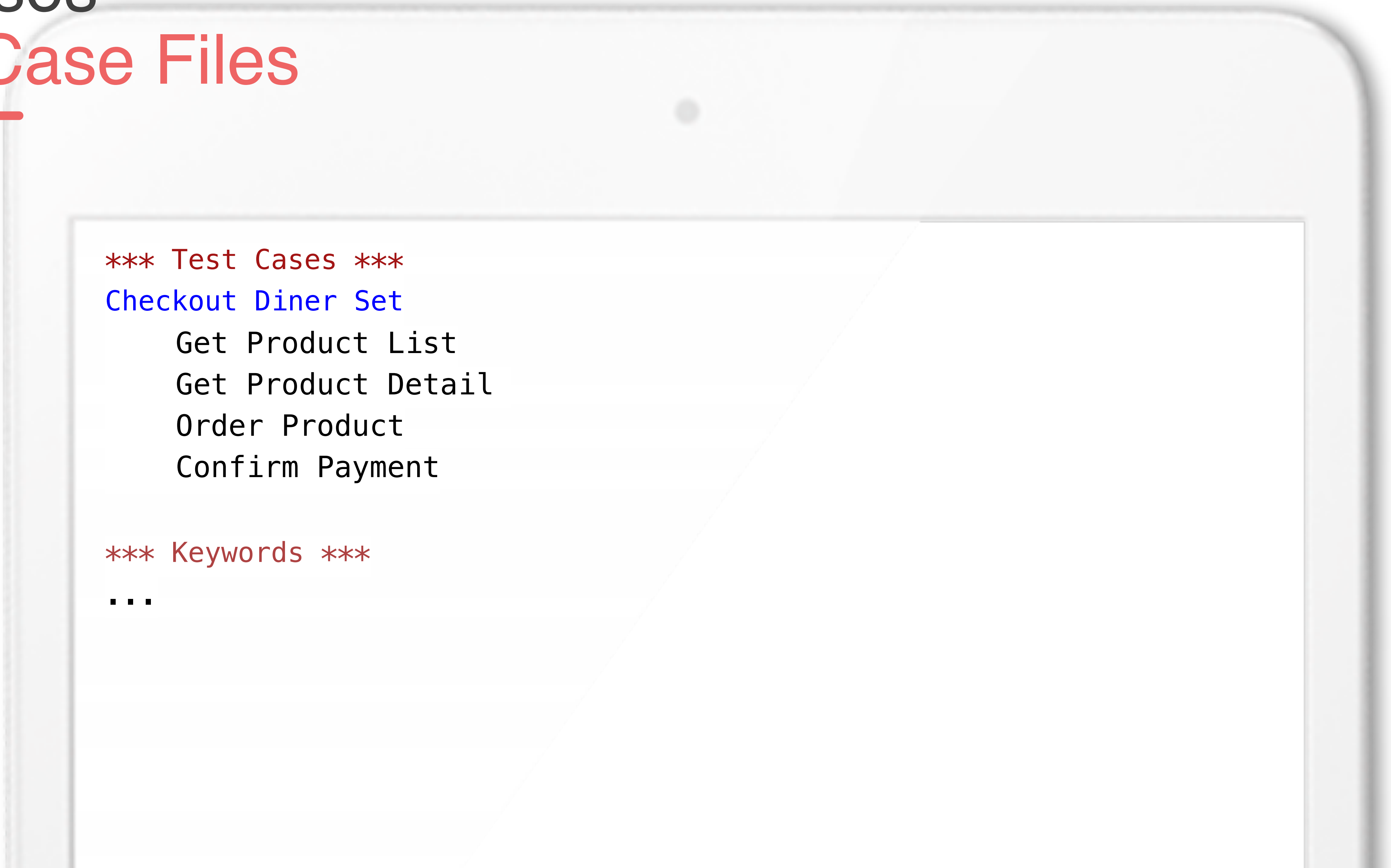
*** Keywords ***

[Arguments]

lean more

© 2020 Siam Chamnankit Company Limited

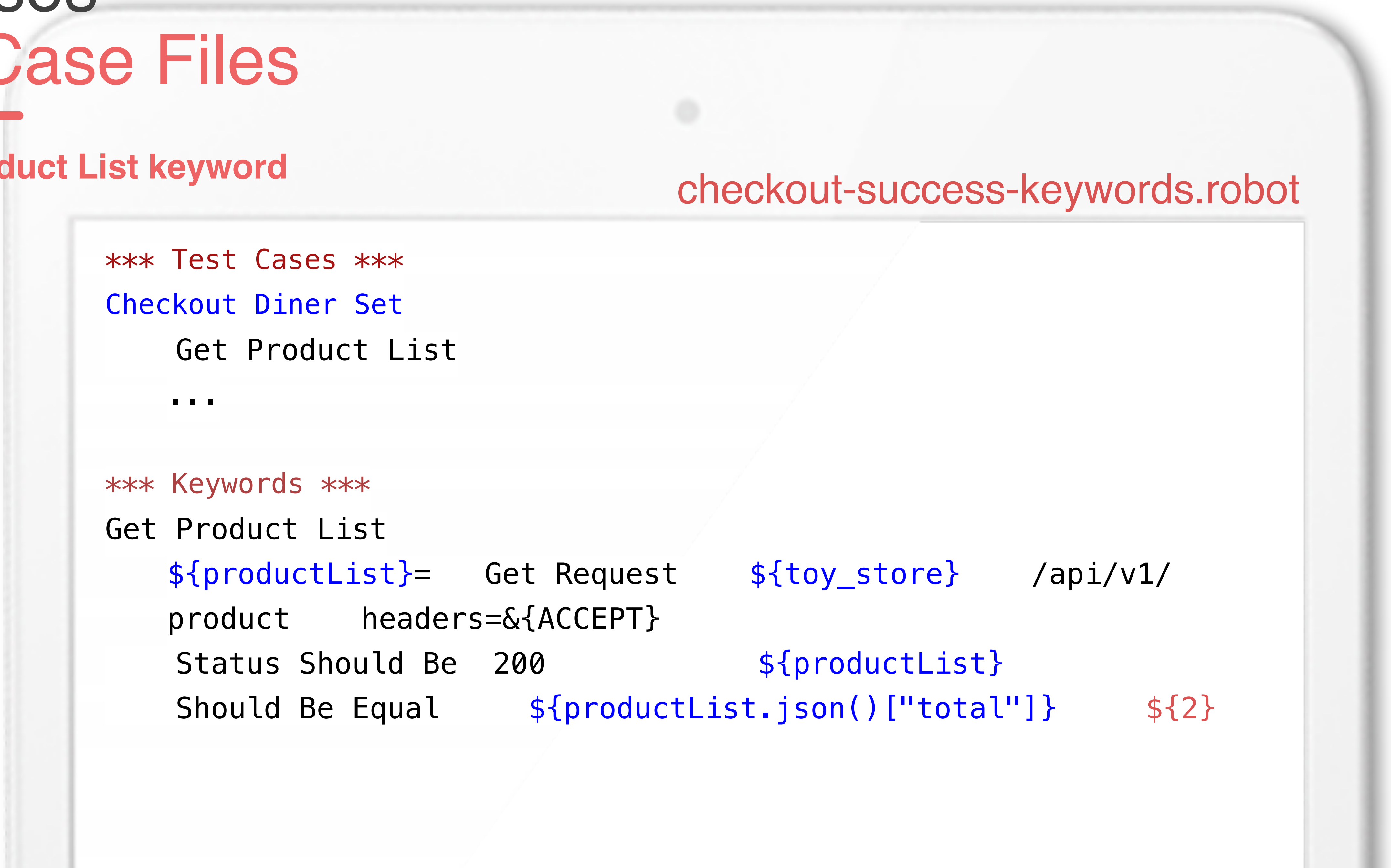
Test Cases
in Test Case Files



```
*** Test Cases ***
Checkout Diner Set
    Get Product List
    Get Product Detail
    Order Product
    Confirm Payment

*** Keywords ***
...
```

Test Cases
in Test Case Files

create Get Product List keyword

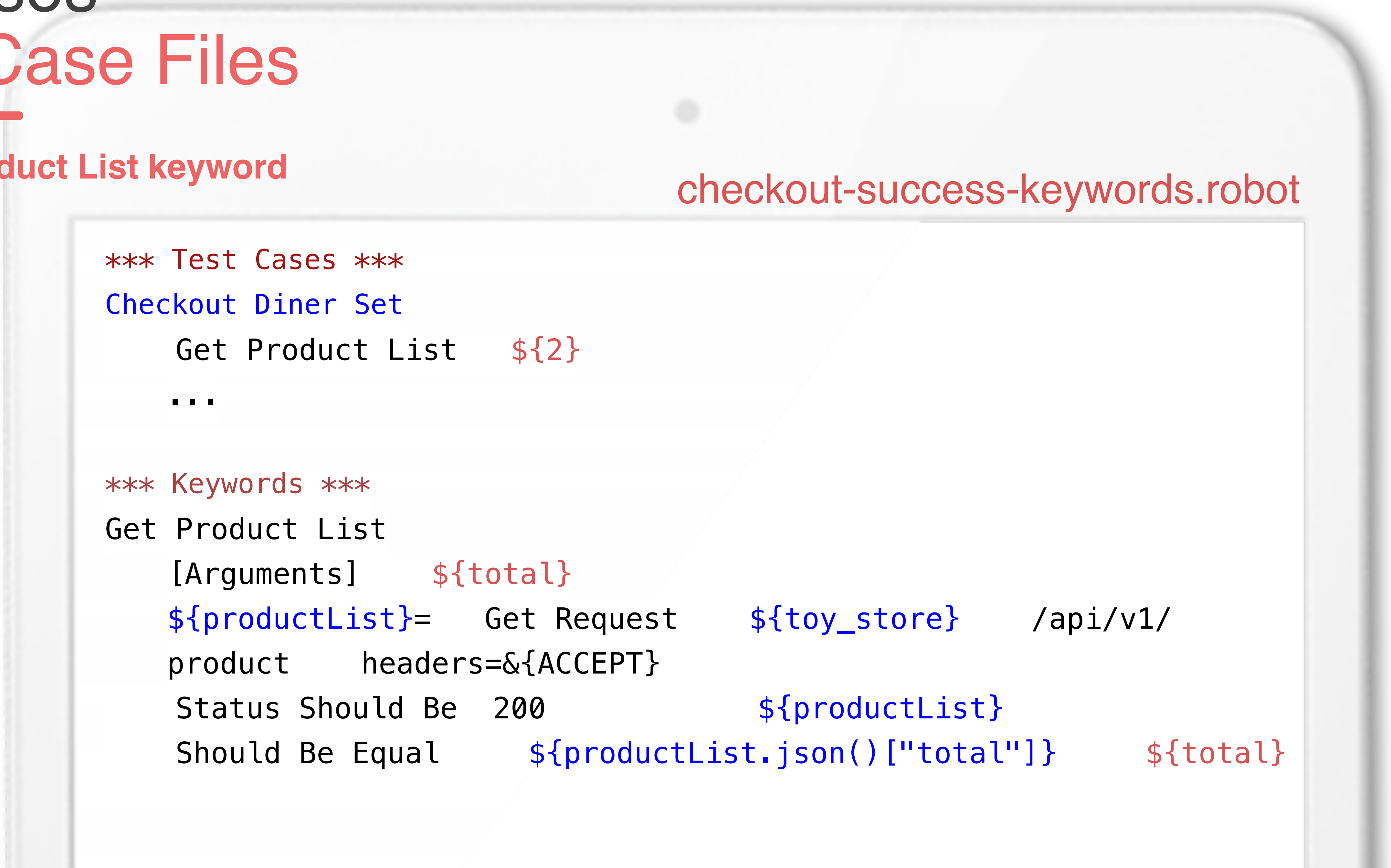


checkout-success-keywords.robot

```
*** Test Cases ***
Checkout Diner Set
     Get Product List
     ...

*** Keywords ***
Get Product List
    ${productList}=   Get Request    ${toy_store}    /api/v1/
    product    headers=&{ACCEPT}
     Status Should Be  200              ${productList}
     Should Be Equal     ${productList.json()["total"]}     ${2}
```

Test Cases
in Test Case Files

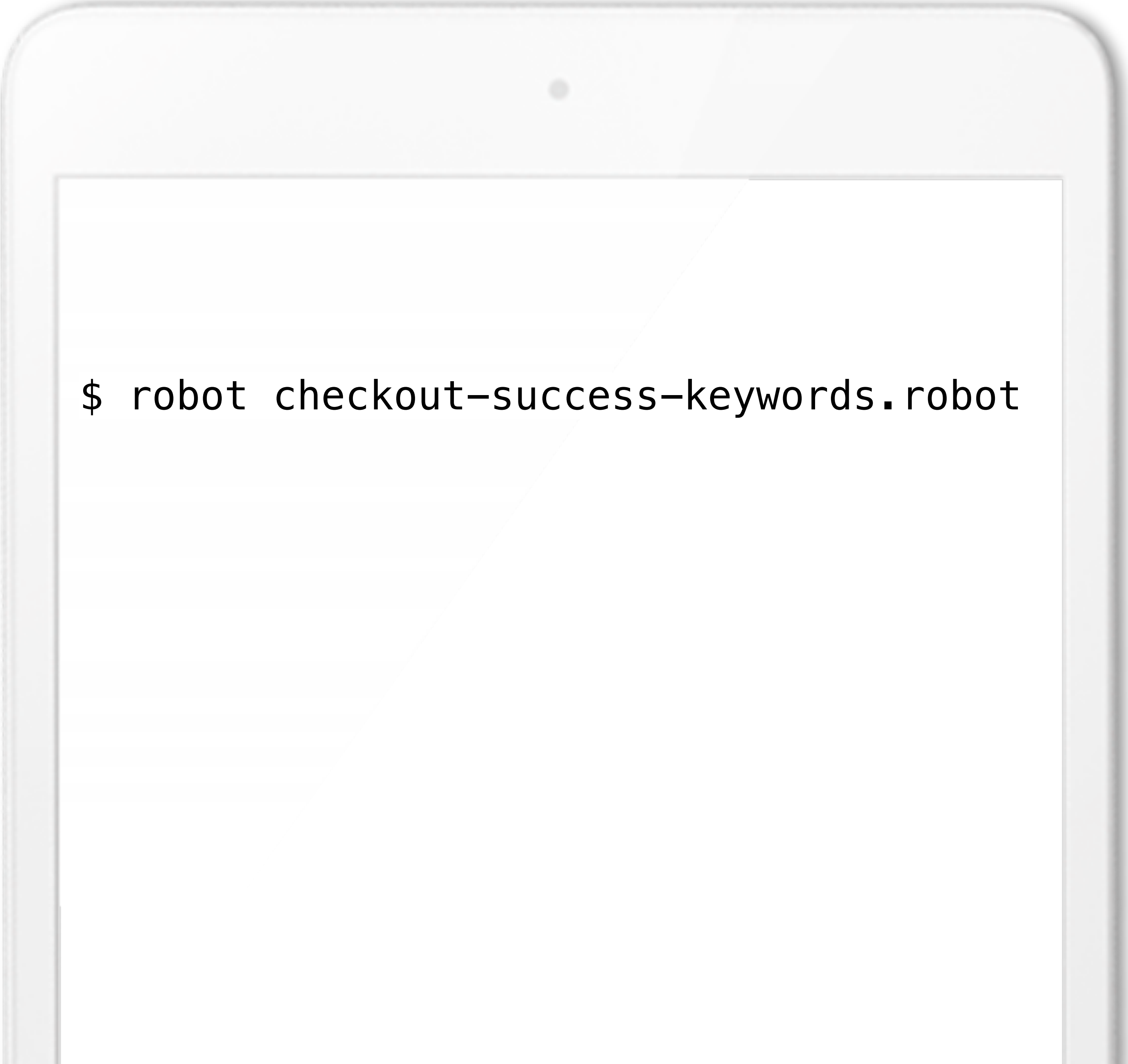
create Get Product List keyword



checkout-success-keywords.robot

```
*** Test Cases ***
Checkout Diner Set
    Get Product List    ${2}
    ...

*** Keywords ***
Get Product List
    [Arguments]    ${total}
    ${productList}=    Get Request    ${toy_store}    /api/v1/
    product    headers=&{ACCEPT}
    Status Should Be  200           ${productList}
    Should Be Equal    ${productList.json()["total"]}    ${total}
```

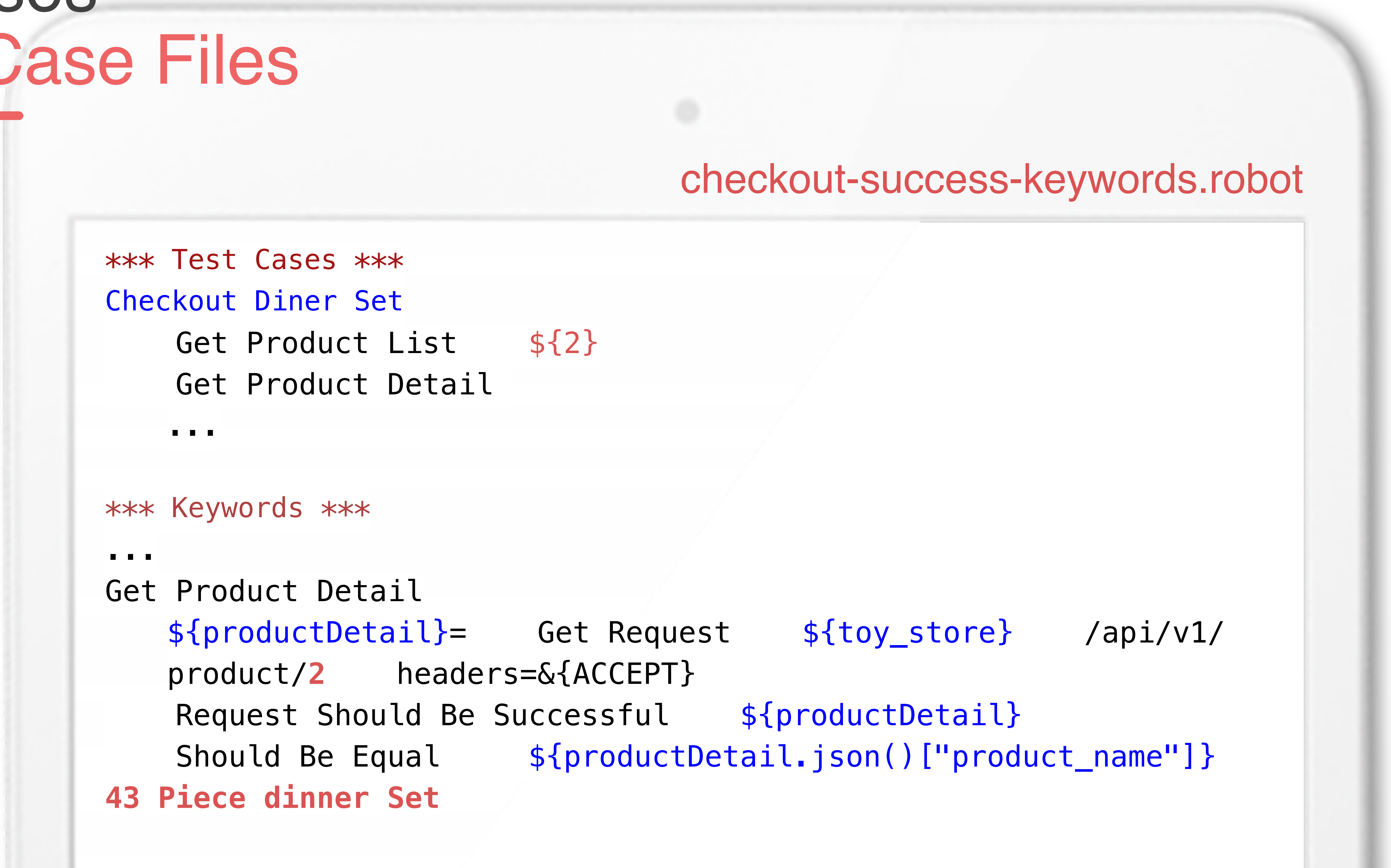
Run
Test Case Files



```
$ robot checkout-success-keywords.robot
```

Test Cases
in Test Case Files

Hard Code!

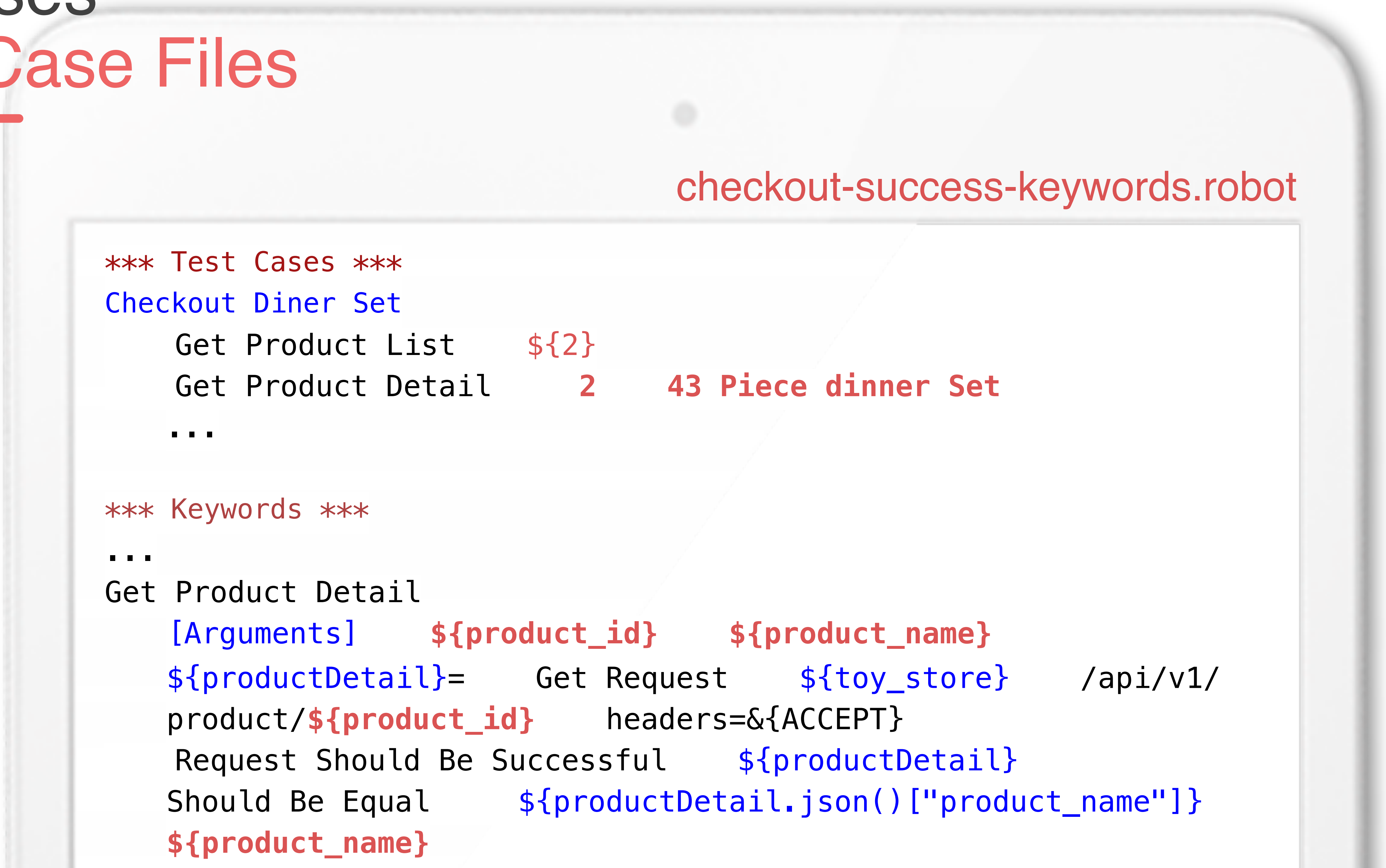


checkout-success-keywords.robot

```
*** Test Cases ***
Checkout Diner Set
    Get Product List    ${2}
    Get Product Detail
    ...

*** Keywords ***
...
Get Product Detail
    ${productDetail}=    Get Request    ${toy_store}    /api/v1/
    product/2    headers=&{ACCEPT}
    Request Should Be Successful    ${productDetail}
    Should Be Equal    ${productDetail.json()["product_name"]}
43 Piece dinner Set
```

Test Cases in Test Case Files

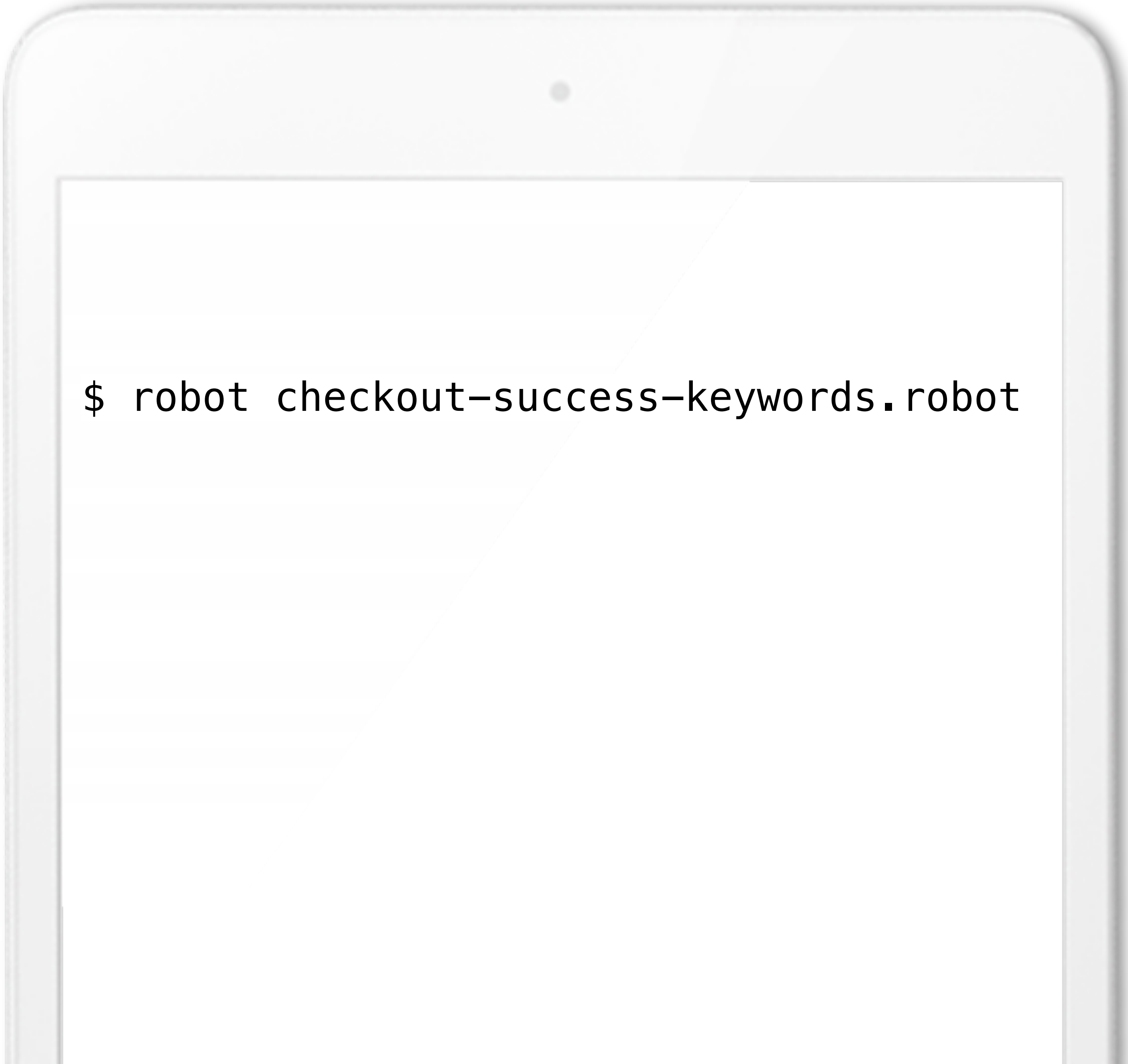
Hard Code!



checkout-success-keywords.robot

```
*** Test Cases ***
Checkout Diner Set
    Get Product List    ${2}
    Get Product Detail    2    43 Piece dinner Set
    ...

*** Keywords ***
...
Get Product Detail
    [Arguments]    ${product_id}    ${product_name}
    ${productDetail}=    Get Request    ${toy_store}    /api/v1/
    product/${product_id}    headers=&{ACCEPT}
    Request Should Be Successful    ${productDetail}
    Should Be Equal    ${productDetail.json()["product_name"]}
    ${product_name}
```

Run
Test Case Files

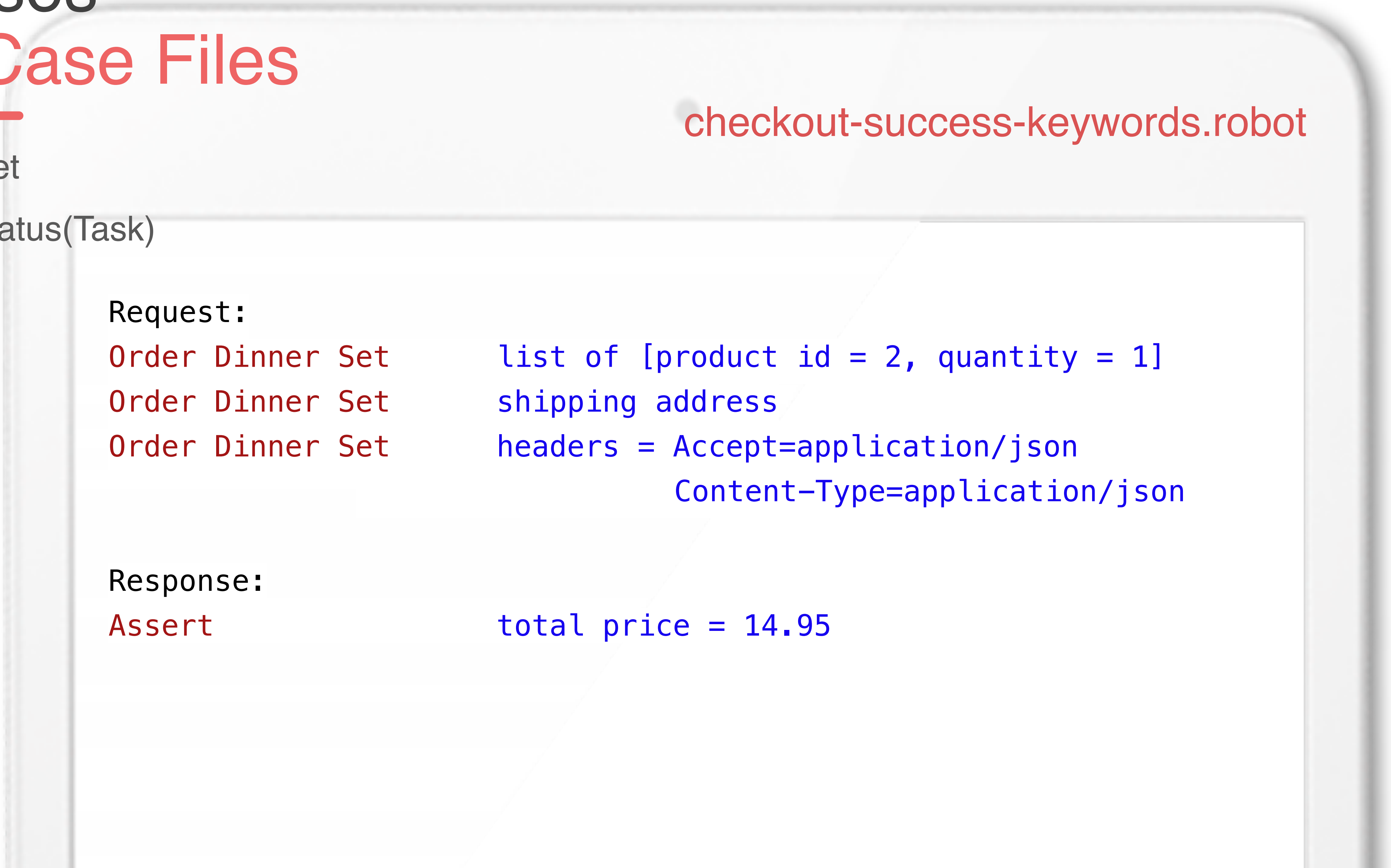


```
$ robot checkout-success-keywords.robot
```

Test Cases
in Test Case Files

Order Dinner Set

Assert Order Status(Task)



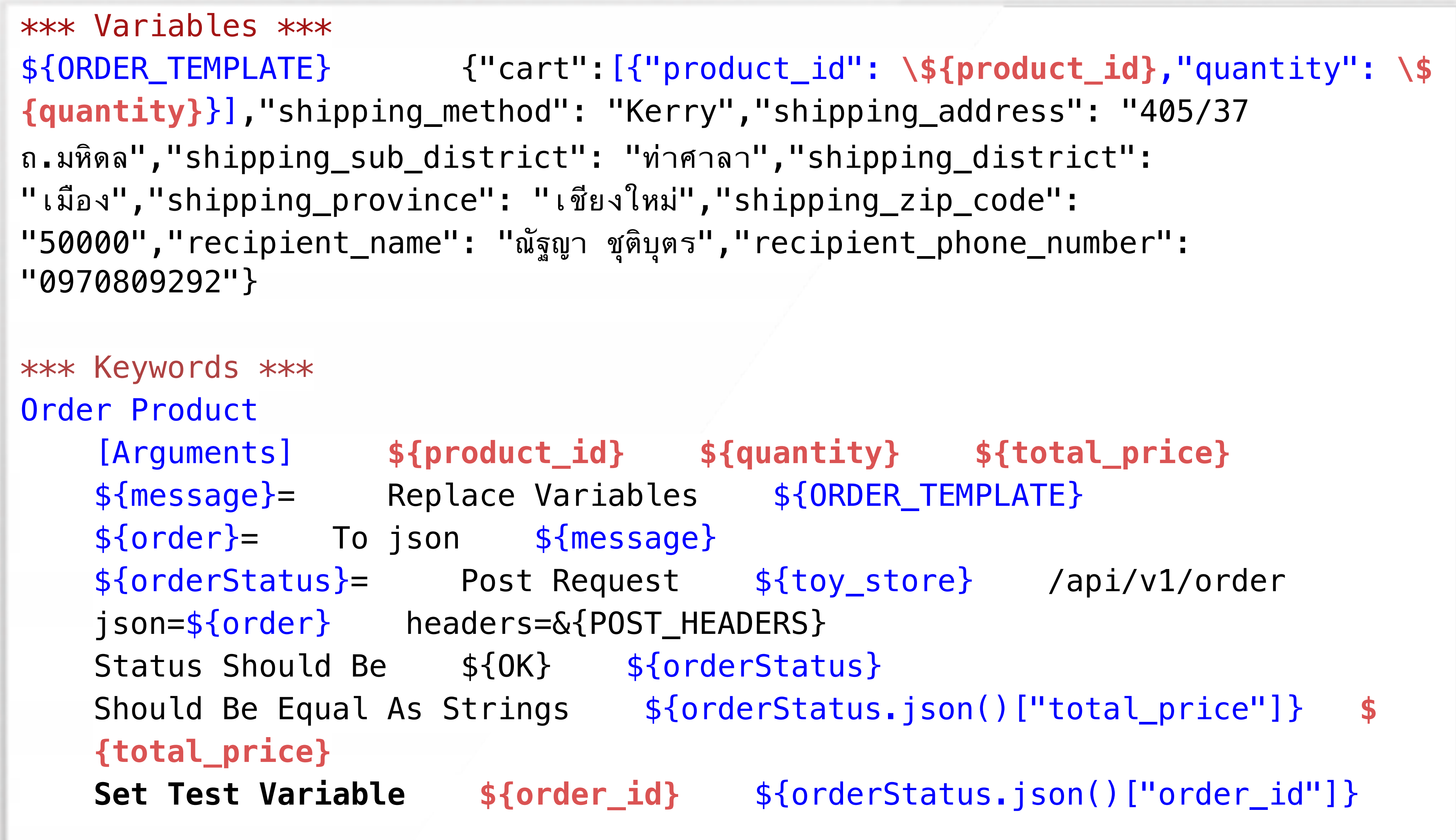
checkout-success-keywords.robot

```
Request:
Order Dinner Set      list of [product id = 2, quantity = 1]
Order Dinner Set      shipping address
Order Dinner Set      headers = Accept=application/json
                                Content-Type=application/json


Response:
Assert                total price = 14.95
```

Test Cases in Test Case Files

Hard Code!

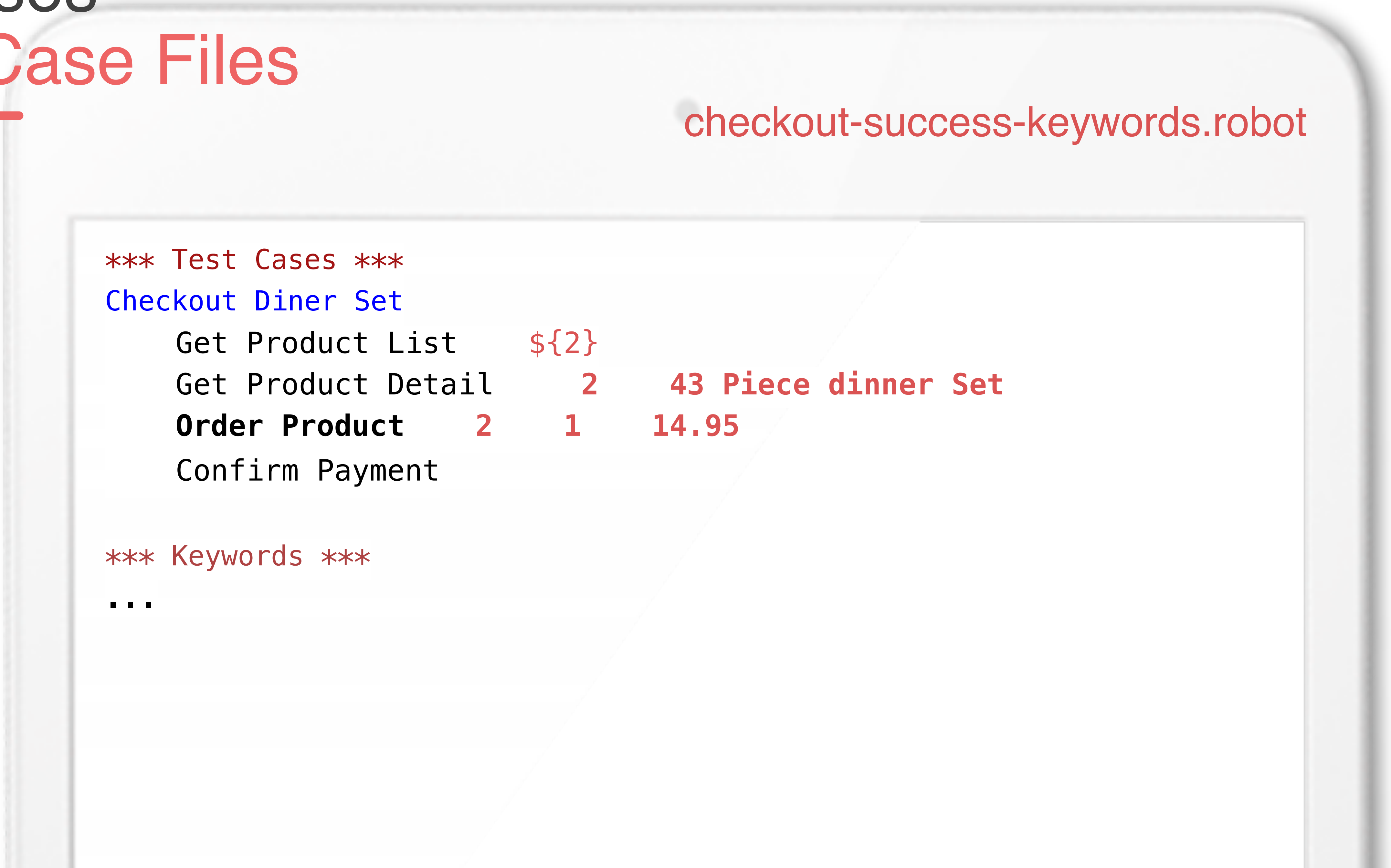
checkout-success-keywords.robot



```
*** Variables ***
${ORDER_TEMPLATE}       {"cart":[{"product_id": \${product_id},"quantity": \$
{quantity}}],"shipping_method": "Kerry","shipping_address": "405/37
ถ.มหิดล","shipping_sub_district": "ท่าศาลา","shipping_district":
"เมือง","shipping_province": "เชียงใหม่","shipping_zip_code":
"50000","recipient_name": "ณัฐญา ชุติบุตร","recipient_phone_number":
"0970809292"}

*** Keywords ***
Order Product
    [Arguments]     ${product_id}    ${quantity}    ${total_price}
    ${message}=     Replace Variables    ${ORDER_TEMPLATE}
    ${order}=    To json    ${message}
    ${orderStatus}=     Post Request    ${toy_store}    /api/v1/order
    json=${order}    headers=&{POST_HEADERS}
    Status Should Be    ${OK}    ${orderStatus}
    Should Be Equal As Strings    ${orderStatus.json()["total_price"]}   $
    {total_price}
    Set Test Variable    ${order_id}    ${orderStatus.json()["order_id"]}
```

Test Cases in Test Case Files

Hard Code!



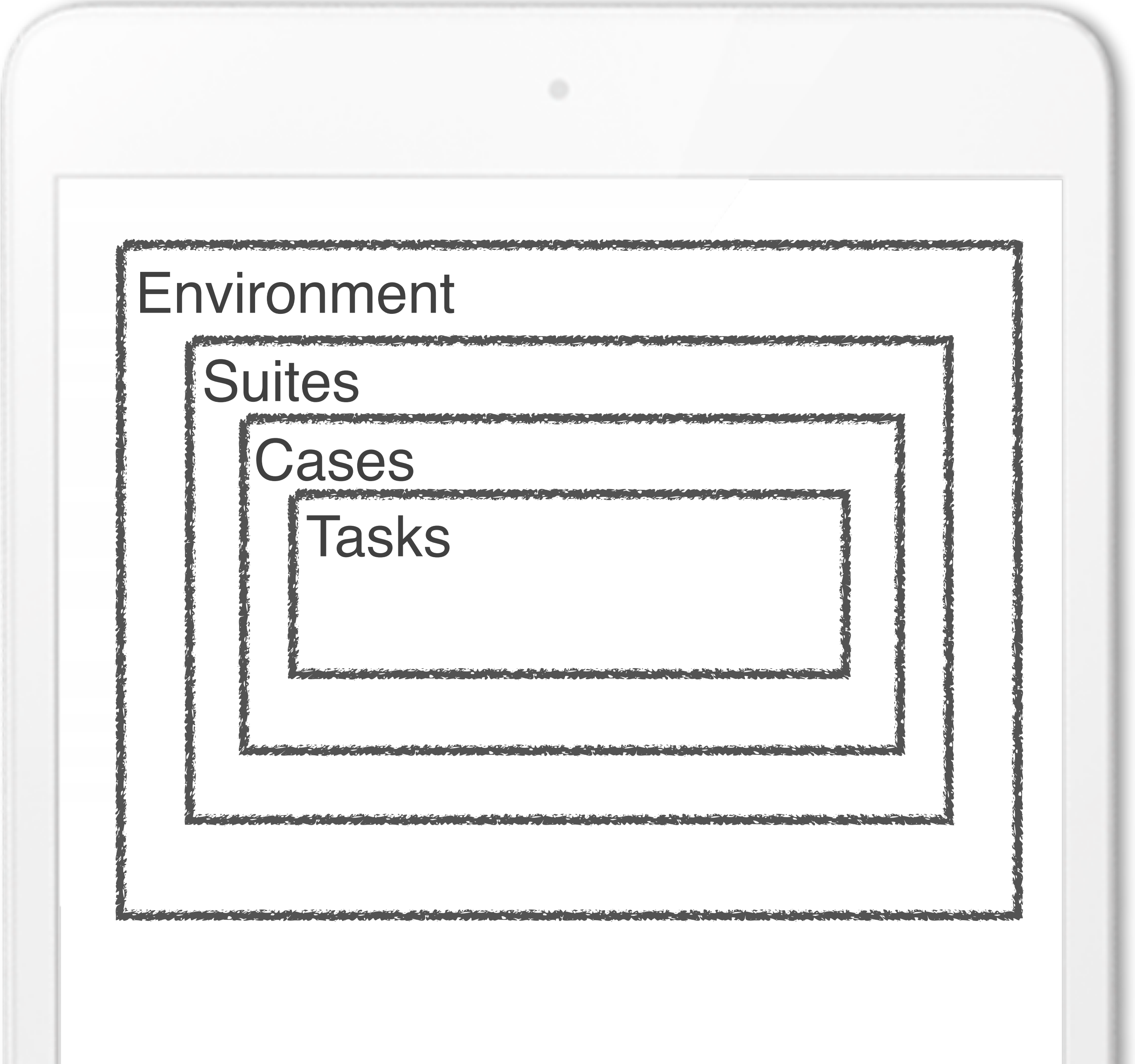
checkout-success-keywords.robot

```
*** Test Cases ***
Checkout Diner Set
    Get Product List    ${2}
    Get Product Detail     2    43 Piece dinner Set
    Order Product    2    1    14.95
    Confirm Payment

*** Keywords ***
...
```

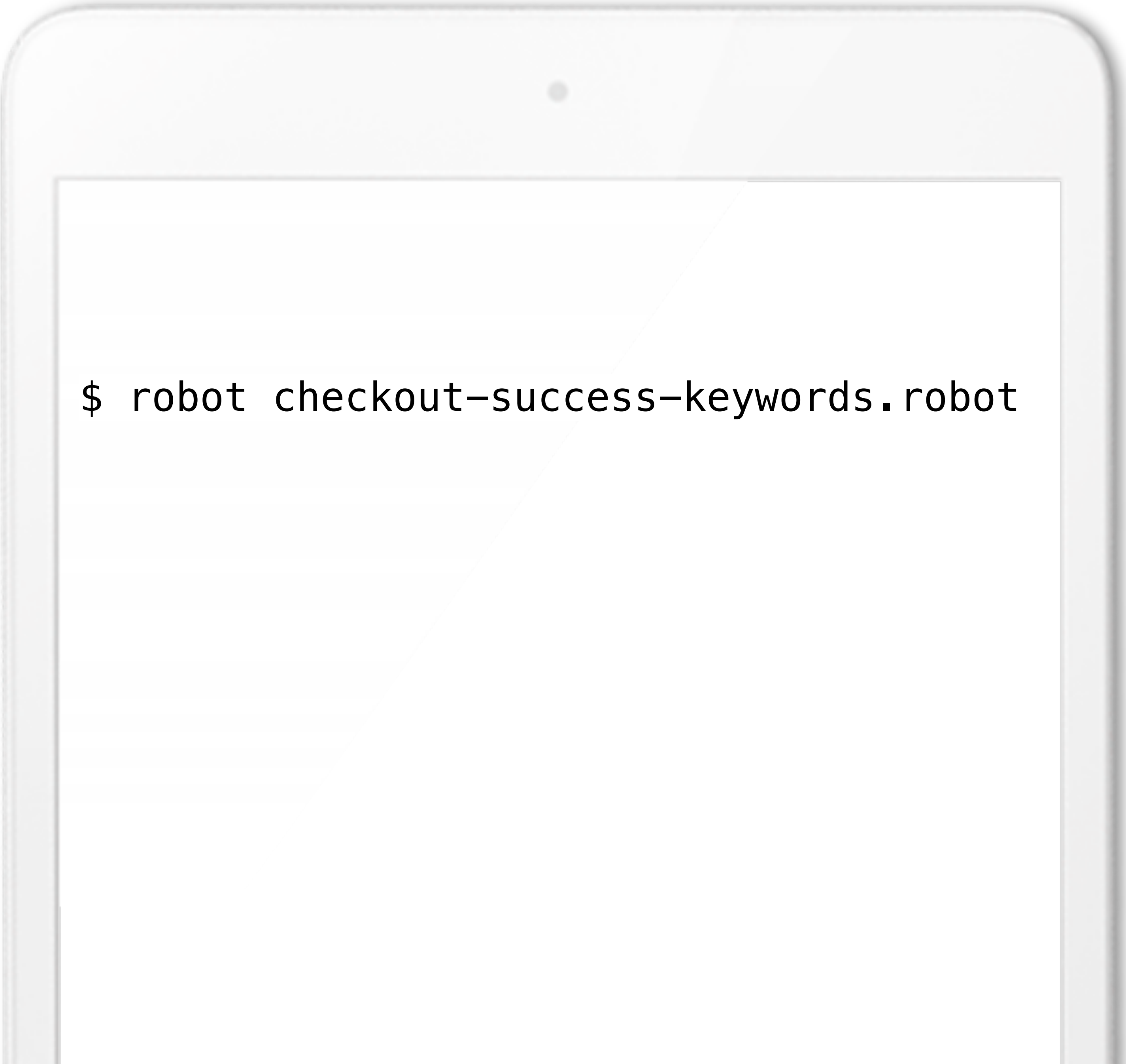
Scope in Test Case Files

Robot Framework data is defined in different sections.

lean more



Run
Test Case Files

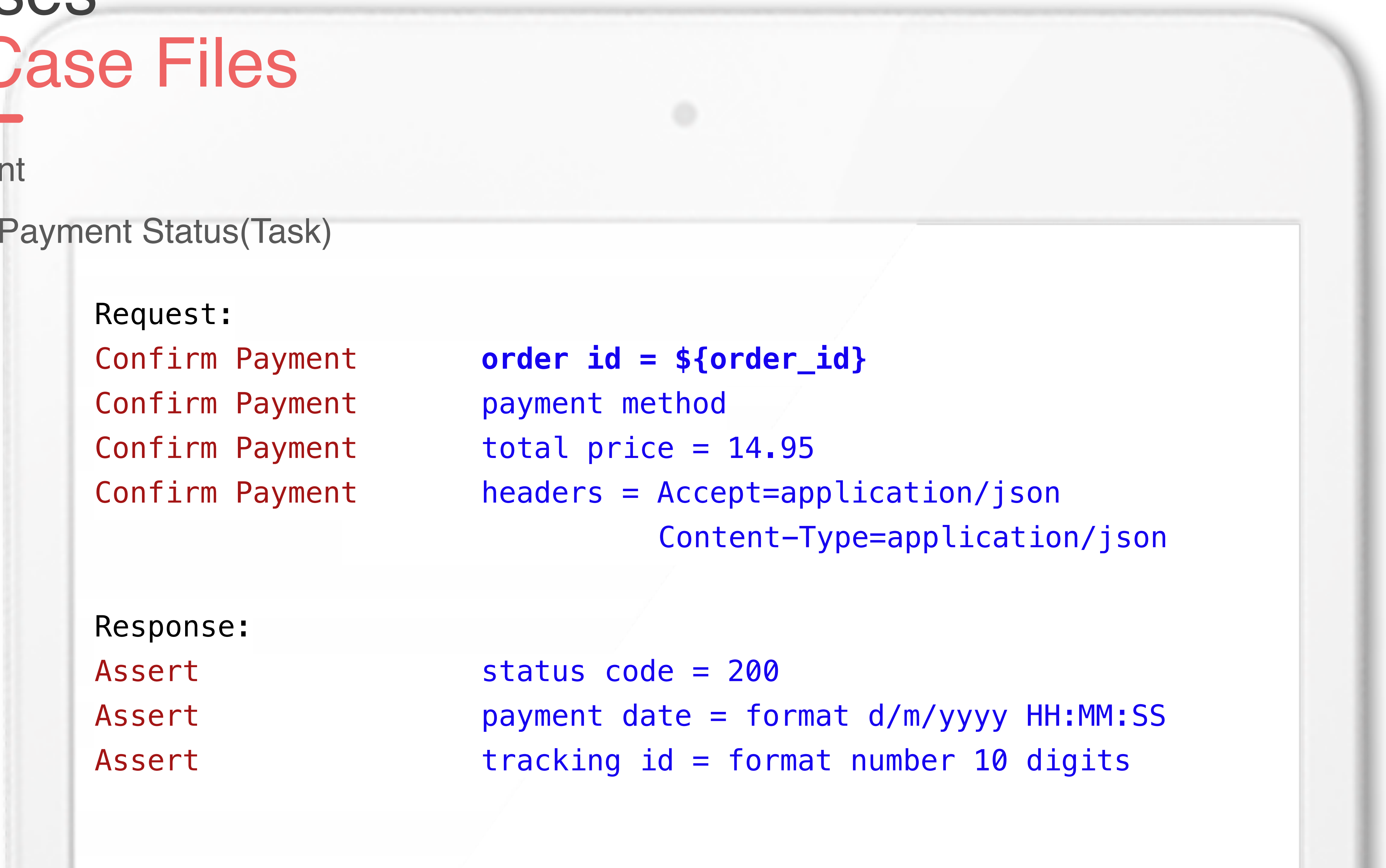


```
$ robot checkout-success-keywords.robot
```

Test Cases
in Test Case Files

Confirm Payment

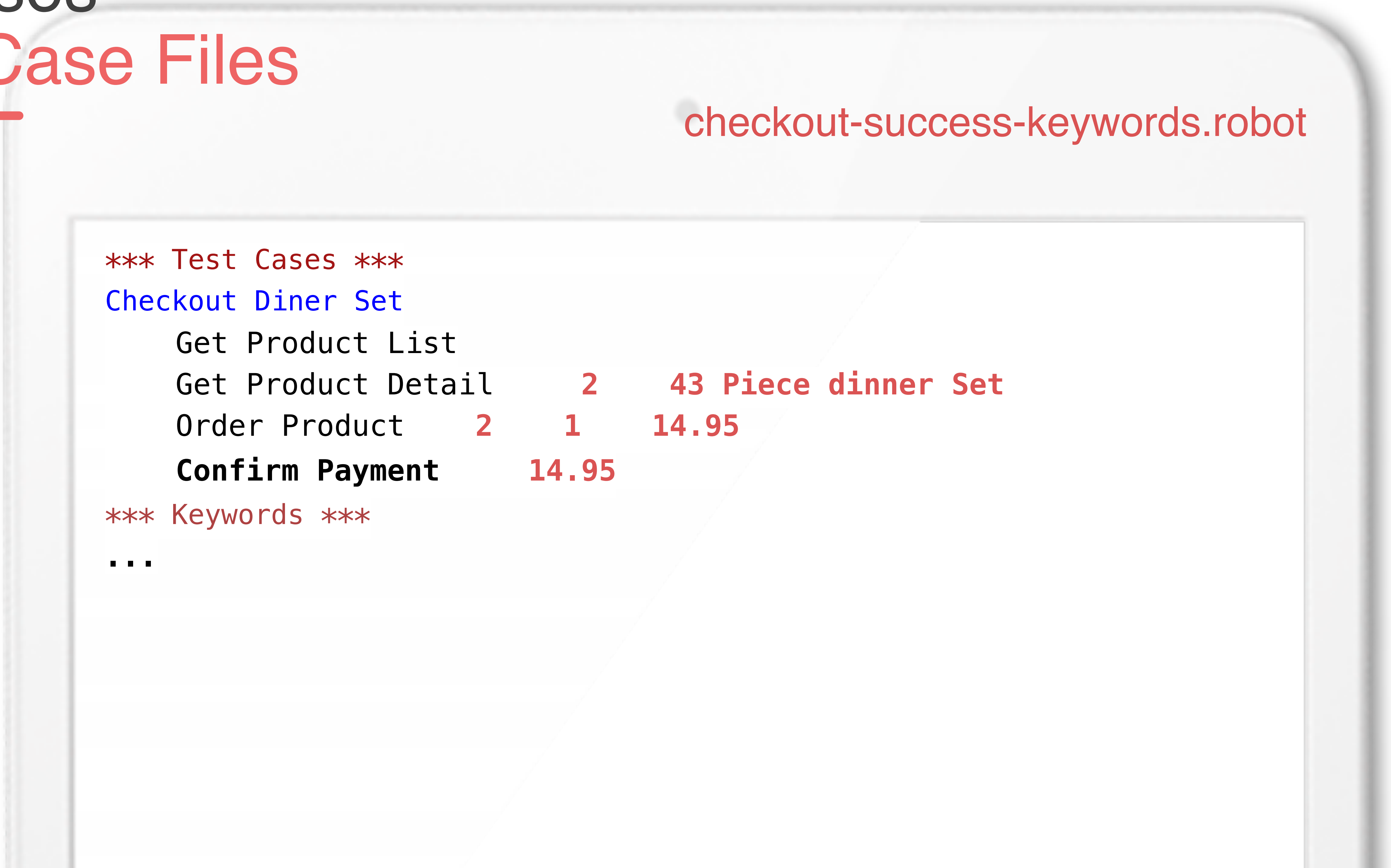
Assert Confirm Payment Status(Task)



```
Request:
Confirm Payment       order id = ${order_id}
Confirm Payment       payment method
Confirm Payment       total price = 14.95
Confirm Payment       headers = Accept=application/json
                                Content-Type=application/json

Response:
Assert                status code = 200
Assert                payment date = format d/m/yyyy HH:MM:SS
Assert                tracking id = format number 10 digits
```

Test Cases in Test Case Files

Hard Code!

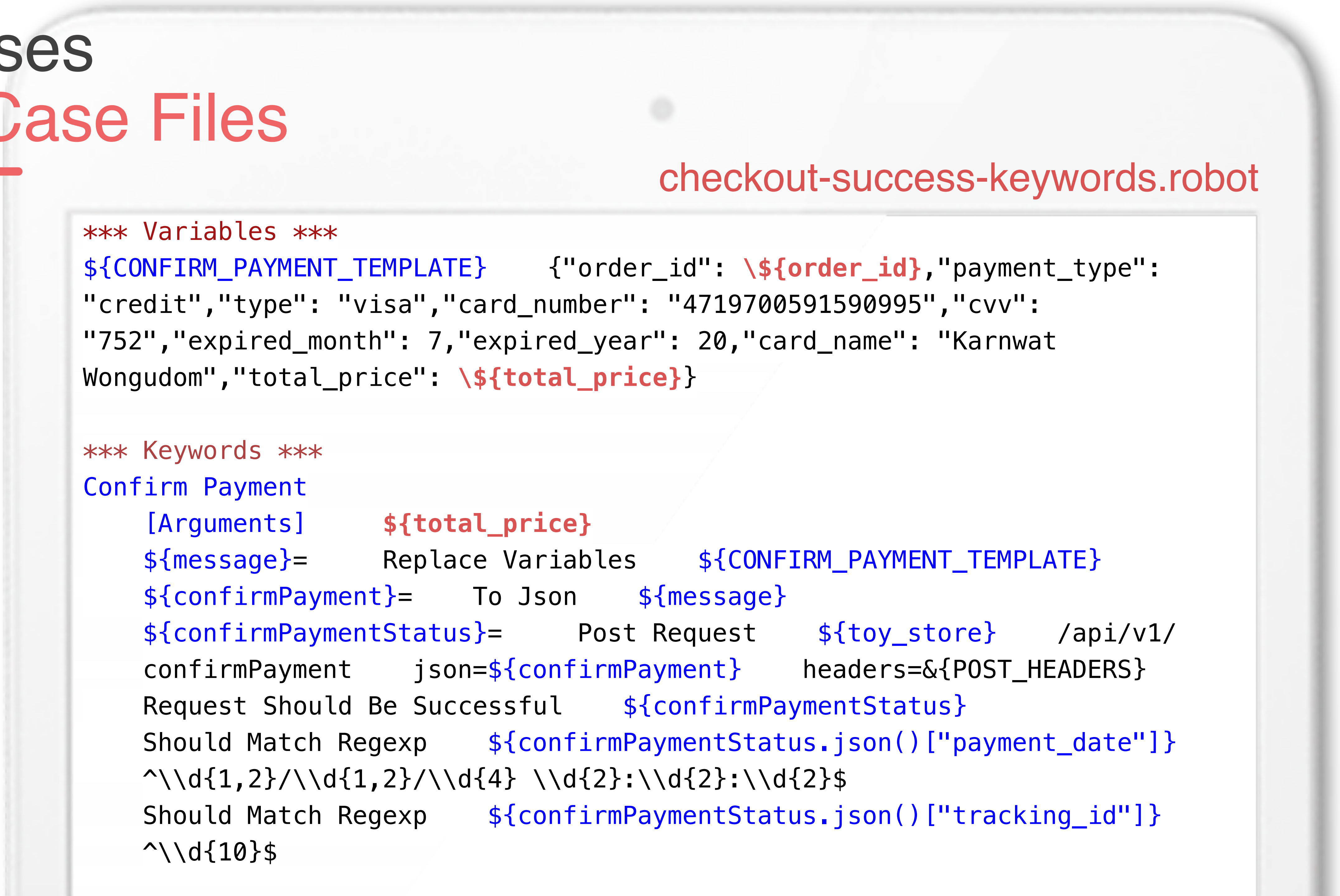


checkout-success-keywords.robot

```
*** Test Cases ***
Checkout Diner Set
    Get Product List
    Get Product Detail     2    43 Piece dinner Set
    Order Product    2    1    14.95
    Confirm Payment     14.95
*** Keywords ***
...
```

Test Cases
in Test Case Files

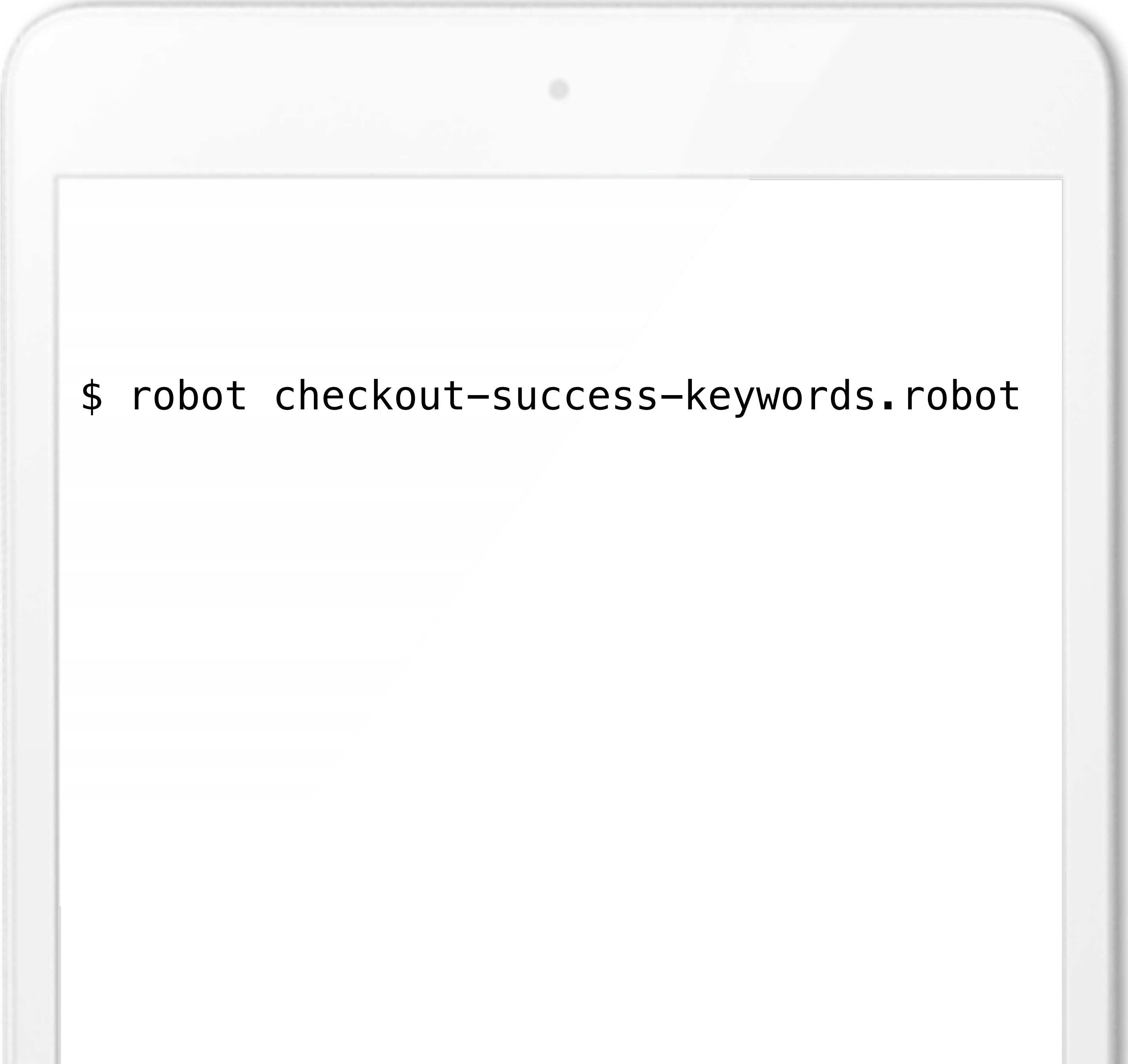
Hard Code!



checkout-success-keywords.robot

```
*** Variables ***
${CONFIRM_PAYMENT_TEMPLATE}    {"order_id": \${order_id},"payment_type":
"credit","type": "visa","card_number": "4719700591590995","cvv":
"752","expired_month": 7,"expired_year": 20,"card_name": "Karnwat
Wongudom","total_price": \${total_price}}

*** Keywords ***
Confirm Payment
    [Arguments]    ${total_price}
    ${message}=    Replace Variables    ${CONFIRM_PAYMENT_TEMPLATE}
    ${confirmPayment}=    To Json    ${message}
    ${confirmPaymentStatus}=    Post Request    ${toy_store}    /api/v1/
    confirmPayment    json=${confirmPayment}    headers=&{POST_HEADERS}
    Request Should Be Successful    ${confirmPaymentStatus}
    Should Match Regexp    ${confirmPaymentStatus.json()["payment_date"]}
    ^\\d{1,2}/\\d{1,2}/\\d{4} \\d{2}:\\d{2}:\\d{2}$
    Should Match Regexp    ${confirmPaymentStatus.json()["tracking_id"]}
    ^\\d{10}$
```

Run
Test Case Files



```
$ robot checkout-success-keywords.robot
```

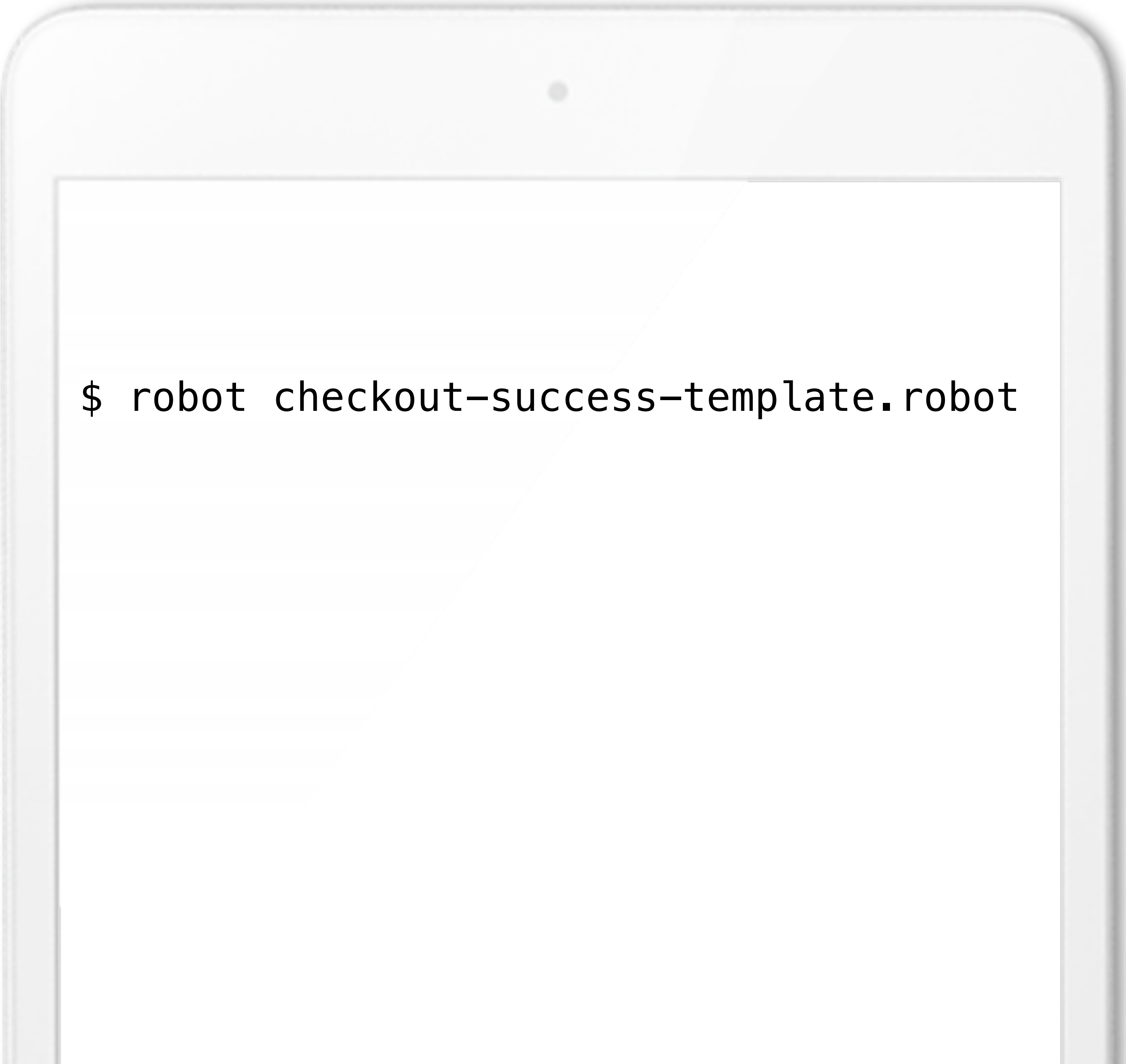
HTTP RequestsLibrary(Python)

*** Template ***

lean more

© 2020 Siam Chamnankit Company Limited

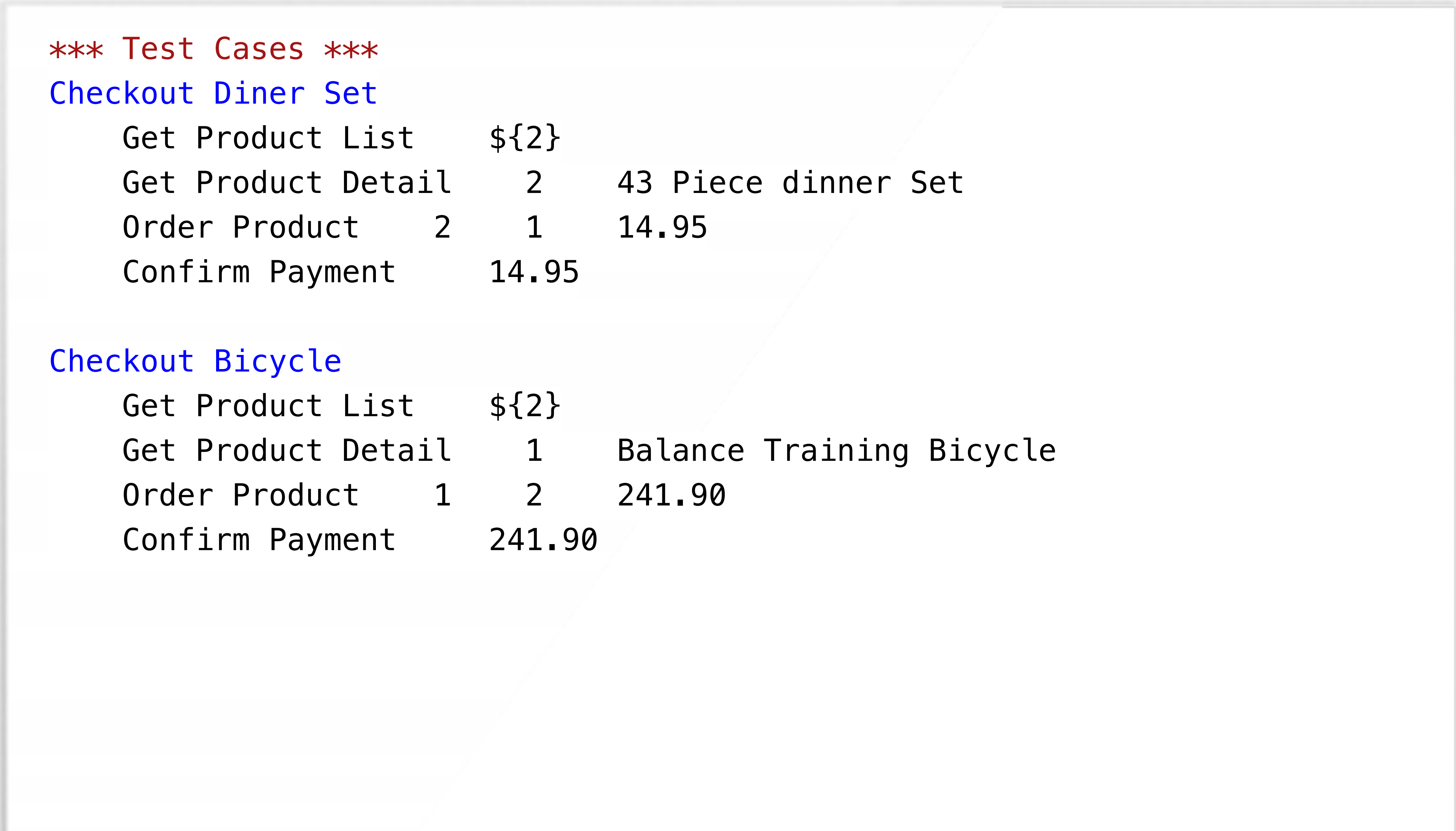
Run
Test Case Files



```
$ robot checkout-success-template.robot
```

Test Cases
in Test Case Files

checkout-success-template.robot

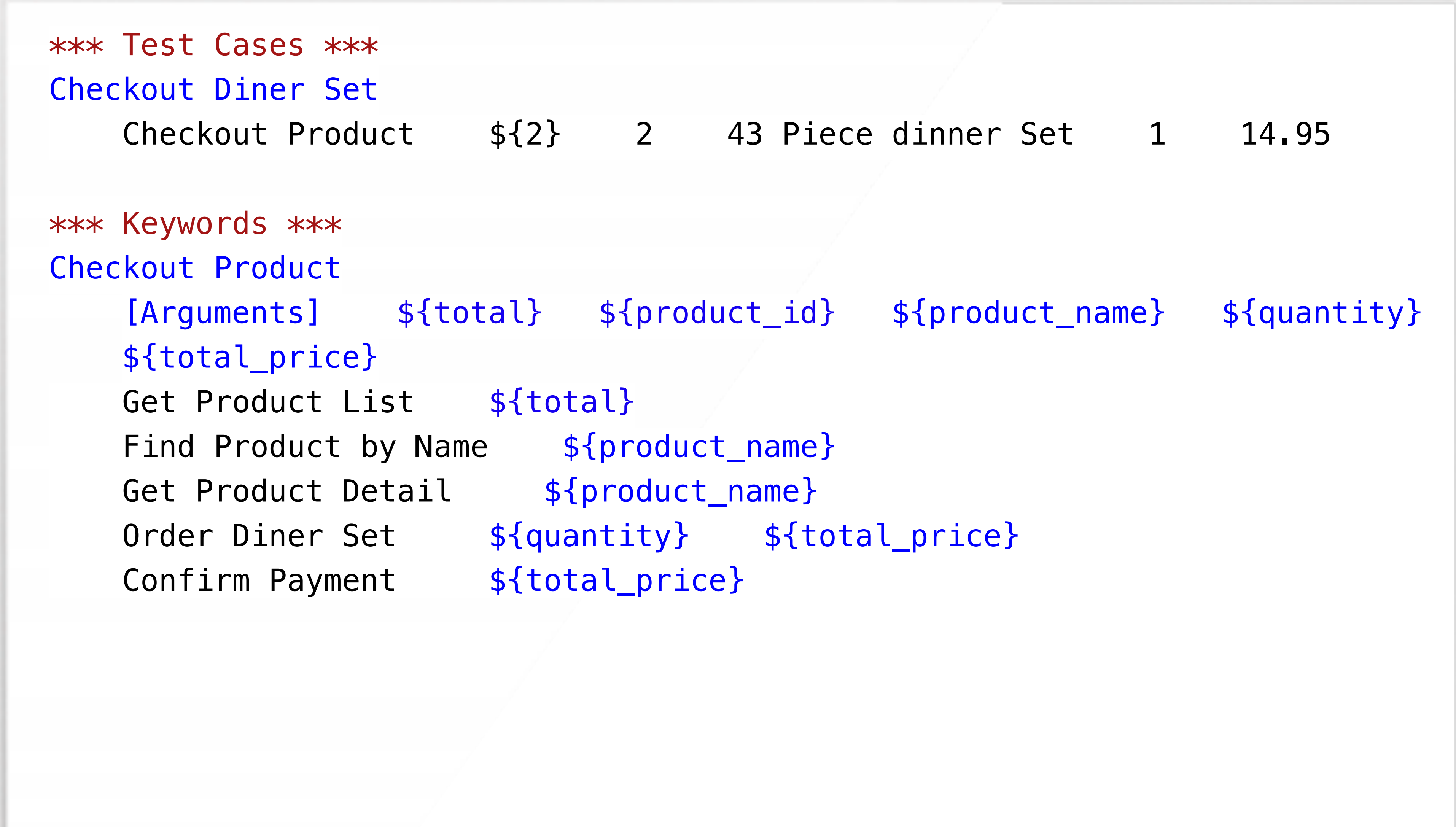


```
*** Test Cases ***
Checkout Diner Set
    Get Product List    ${2}
    Get Product Detail    2    43 Piece dinner Set
    Order Product    2    1    14.95
    Confirm Payment    14.95

Checkout Bicycle
    Get Product List    ${2}
    Get Product Detail    1    Balance Training Bicycle
    Order Product    1    2    241.90
    Confirm Payment    241.90
```

Test Cases
in Test Case Files

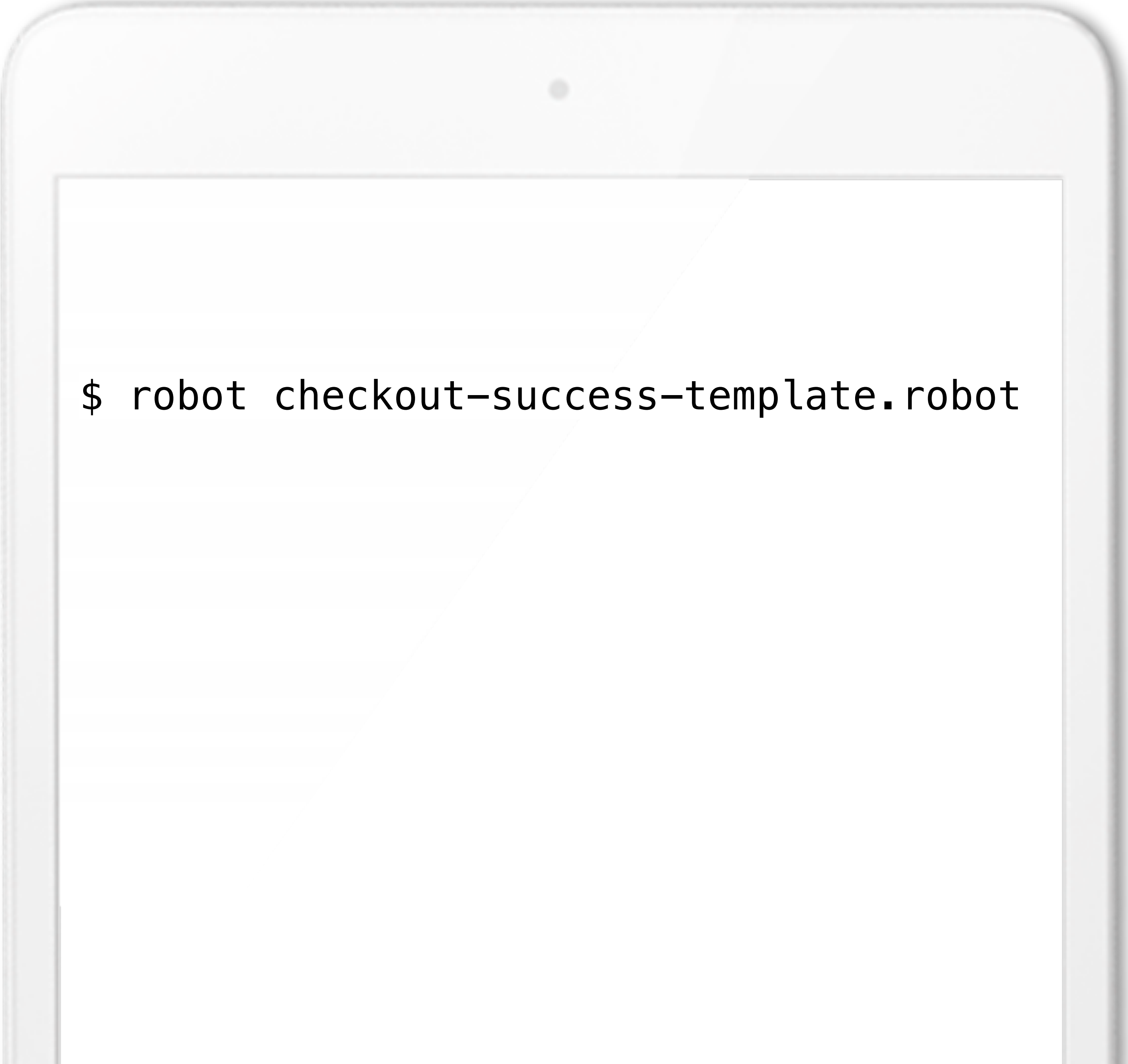
Hard Code!

checkout-success-template.robot



```
*** Test Cases ***
Checkout Diner Set
    Checkout Product    ${2}    2    43 Piece dinner Set    1    14.95

*** Keywords ***
Checkout Product
    [Arguments]    ${total}    ${product_id}    ${product_name}    ${quantity}
    ${total_price}
    Get Product List    ${total}
    Find Product by Name    ${product_name}
    Get Product Detail    ${product_name}
    Order Diner Set    ${quantity}    ${total_price}
    Confirm Payment    ${total_price}
```

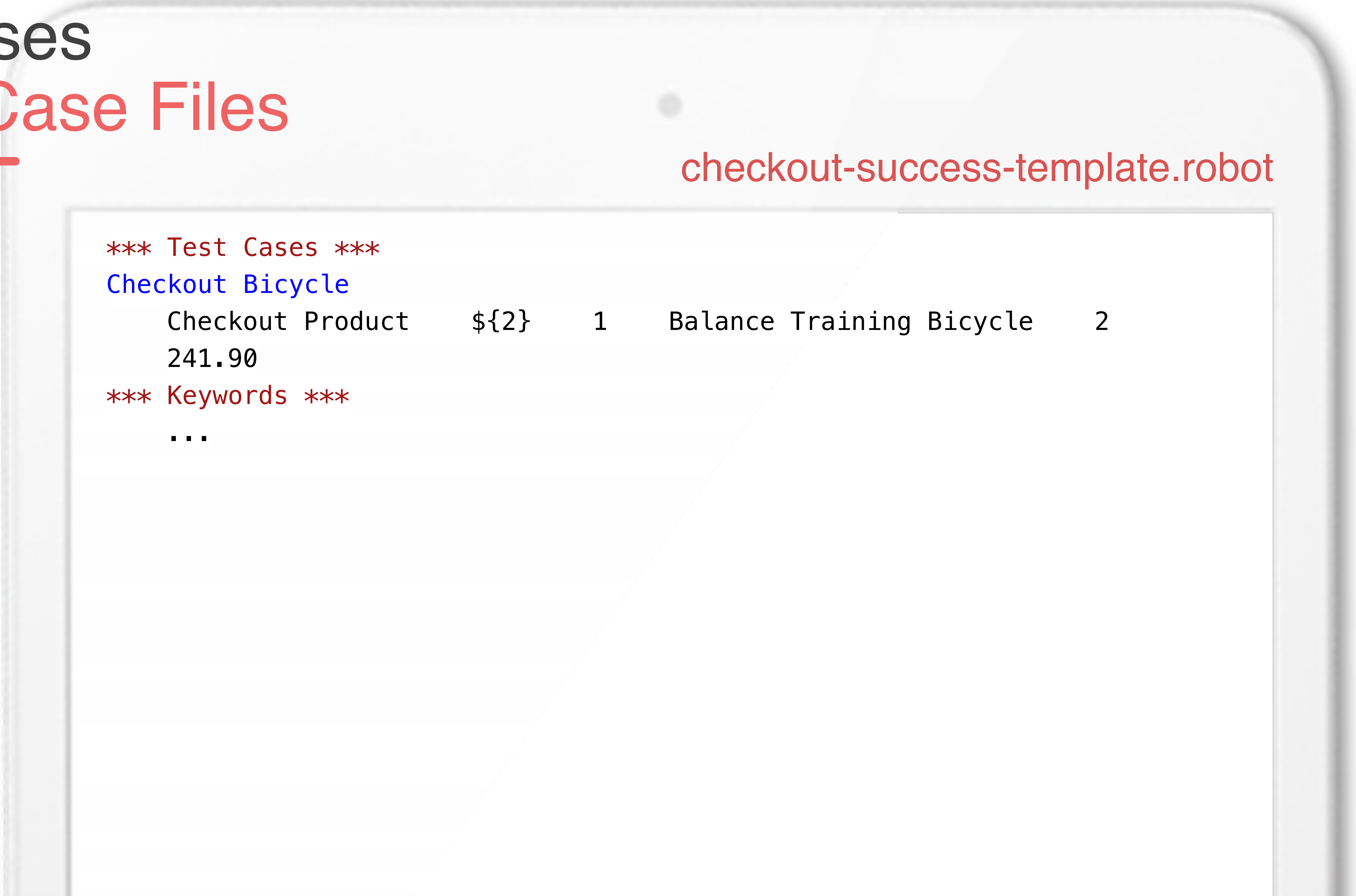
Run
Test Case Files



```
$ robot checkout-success-template.robot
```

Test Cases
in Test Case Files

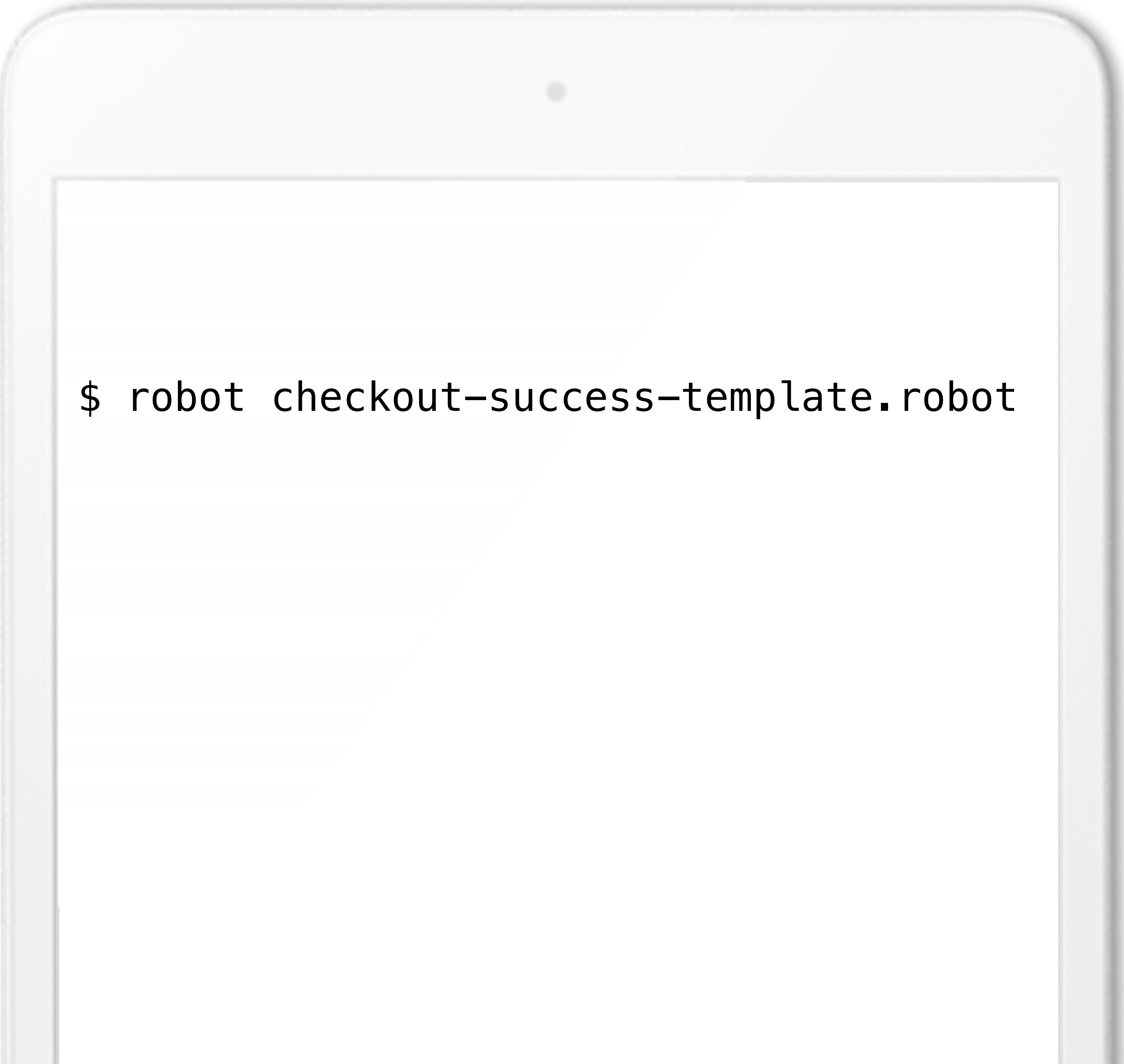
Hard Code!



checkout-success-template.robot

```
*** Test Cases ***
Checkout Bicycle
    Checkout Product    ${2}    1    Balance Training Bicycle    2
    241.90
*** Keywords ***
    ...
```

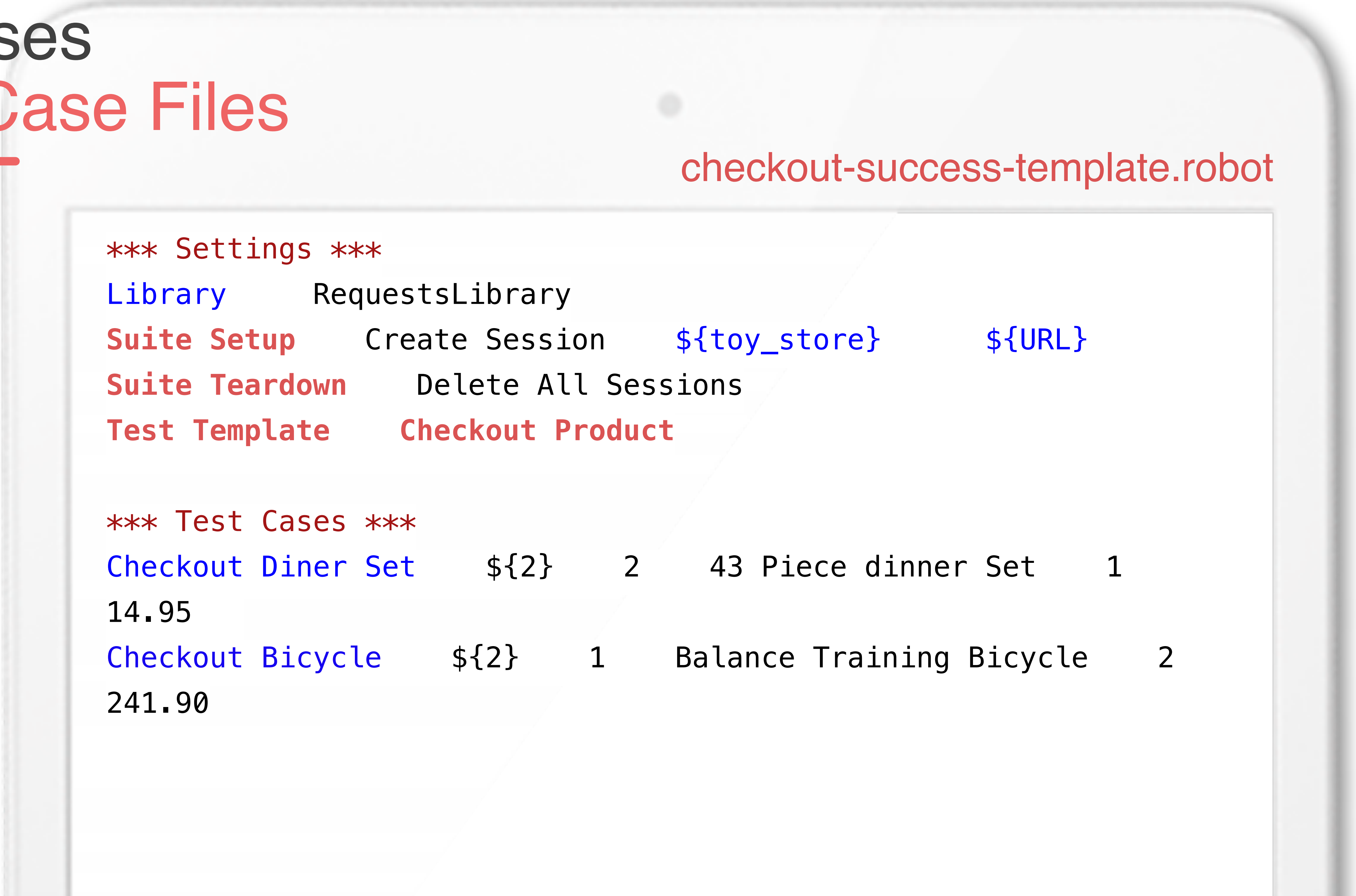
Run
Test Case Files



```
$ robot checkout-success-template.robot
```

Test Cases in Test Case Files

Hard Code!



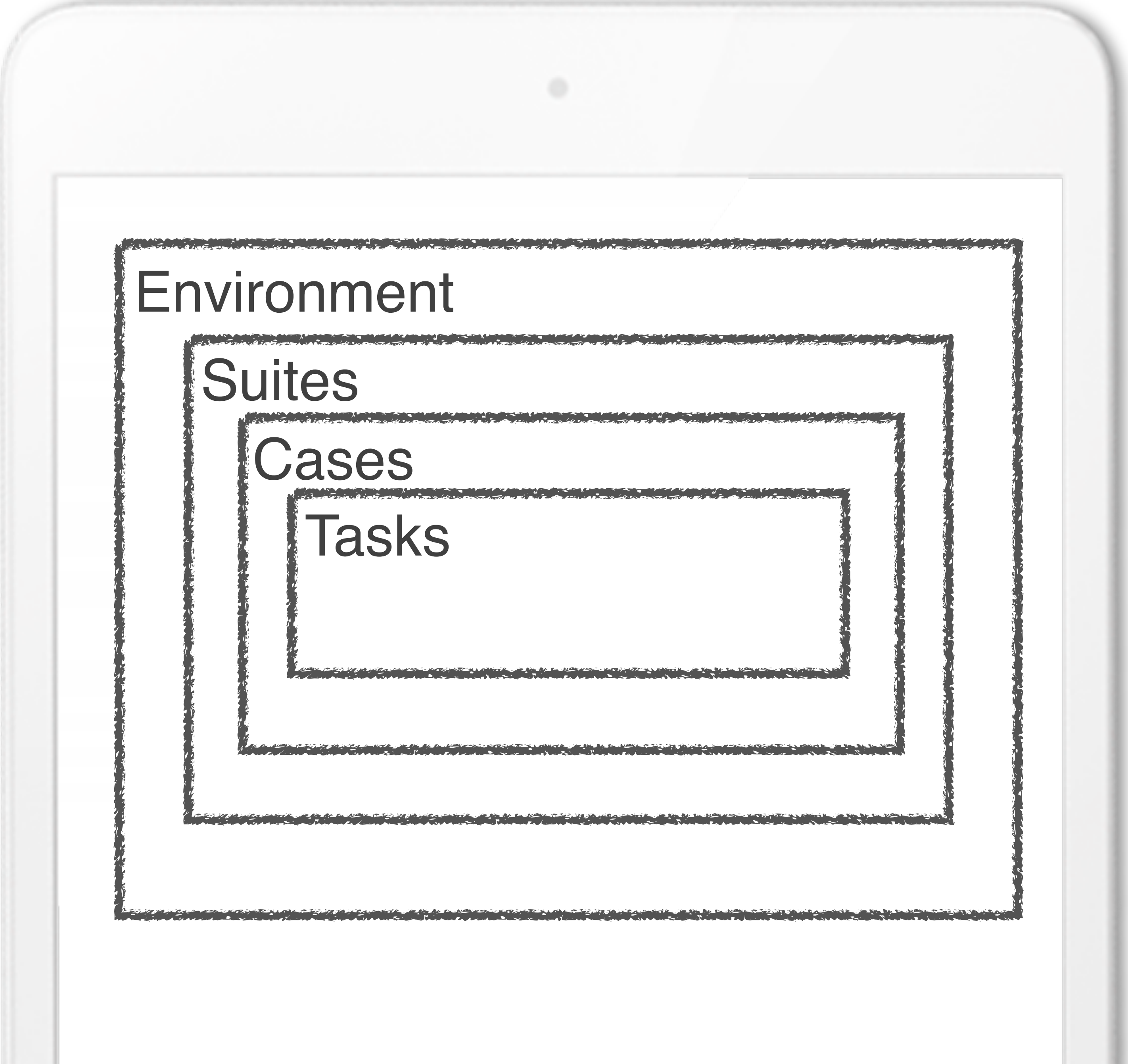
checkout-success-template.robot

```
*** Settings ***
Library        RequestsLibrary
Suite Setup      Create Session     ${toy_store}       ${URL}
Suite Teardown      Delete All Sessions
Test Template      Checkout Product

*** Test Cases ***
Checkout Diner Set     ${2}     2     43 Piece dinner Set     1
14.95
Checkout Bicycle     ${2}     1     Balance Training Bicycle     2
241.90
```

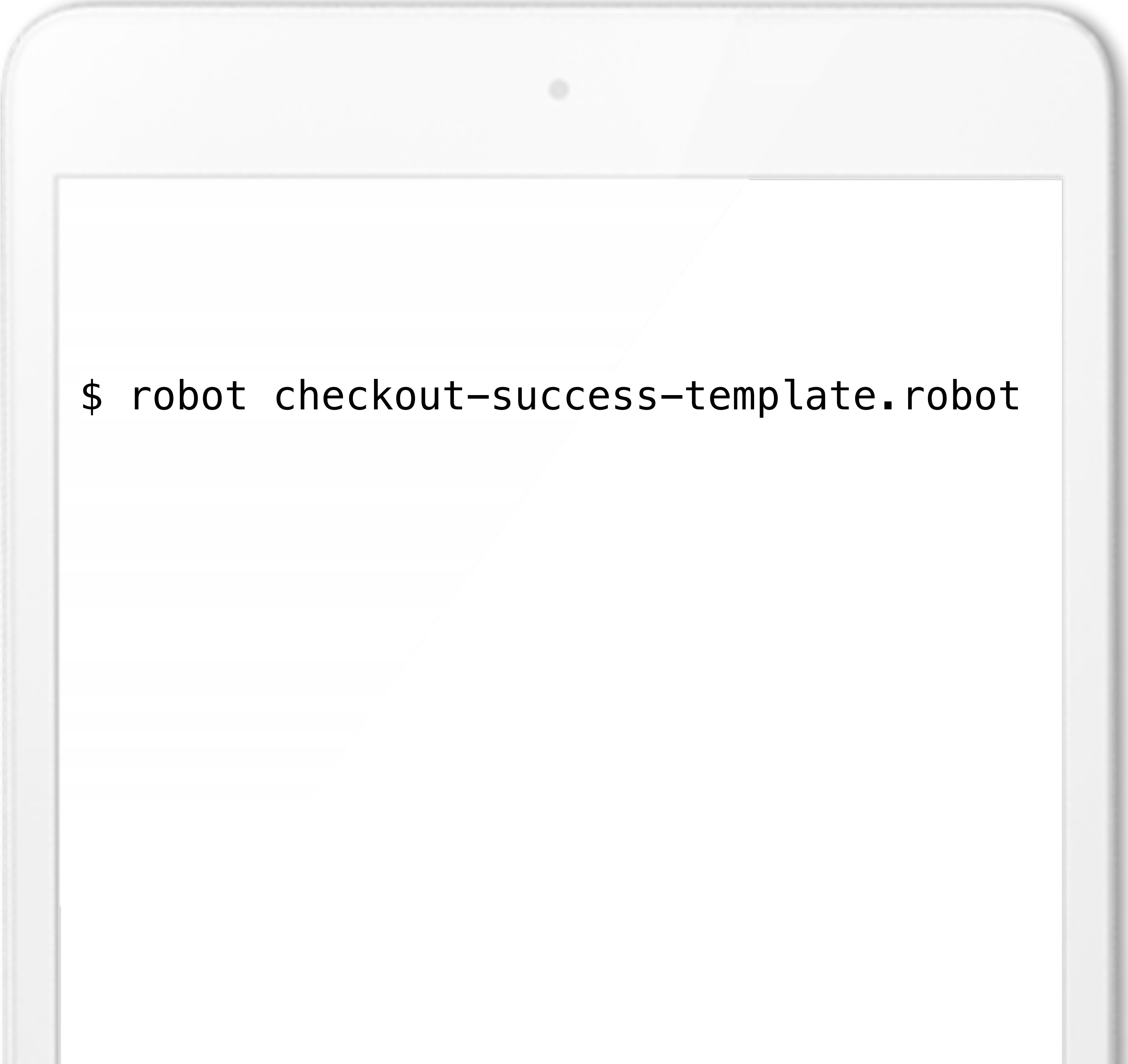
Scope in Test Case Files

Robot Framework data is defined in different sections.

lean more



Run
Test Case Files



```
$ robot checkout-success-template.robot
```

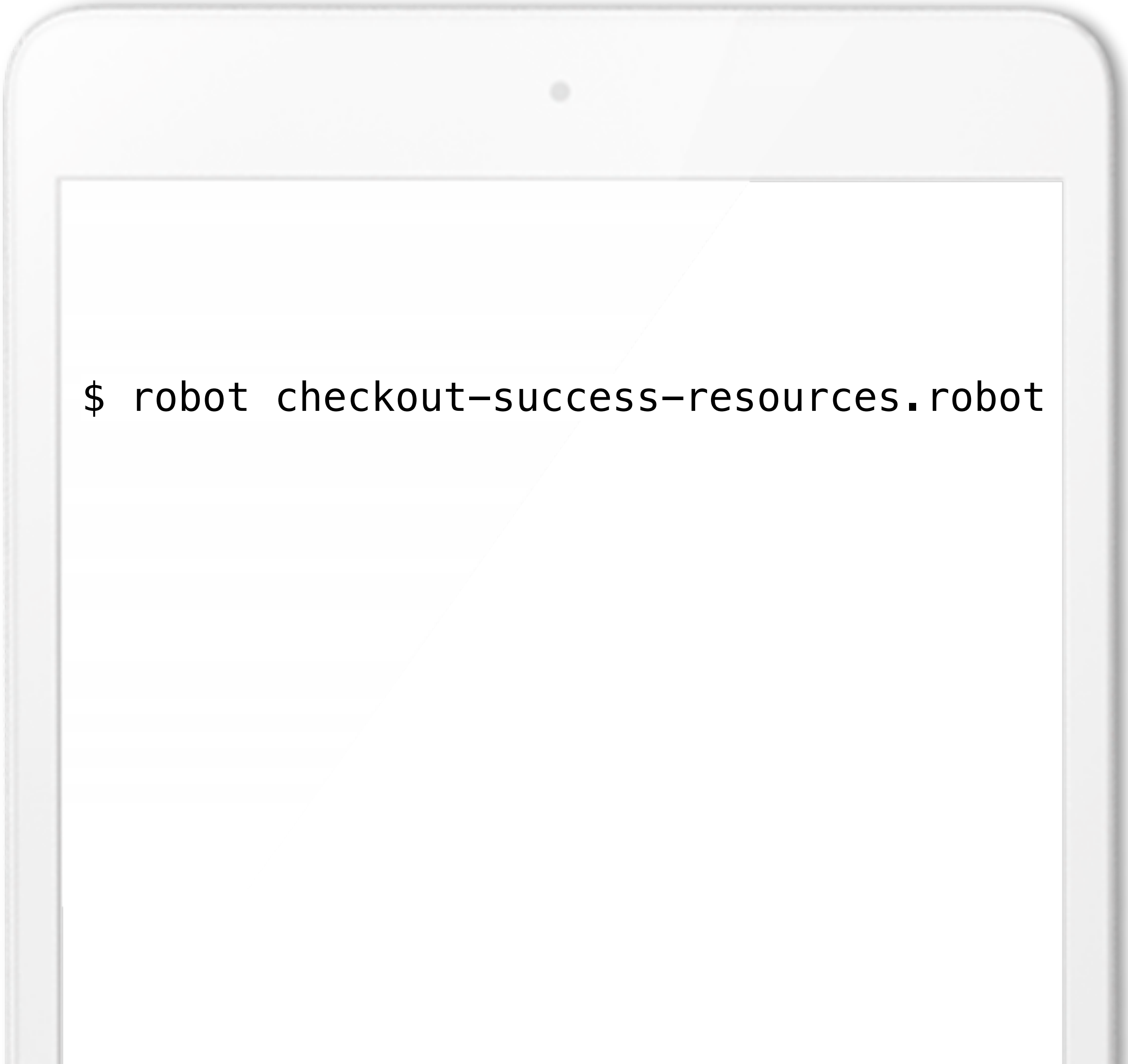
HTTP RequestsLibrary(Python)

*** Resource ***

lean more

© 2020 Siam Chamnankit Company Limited

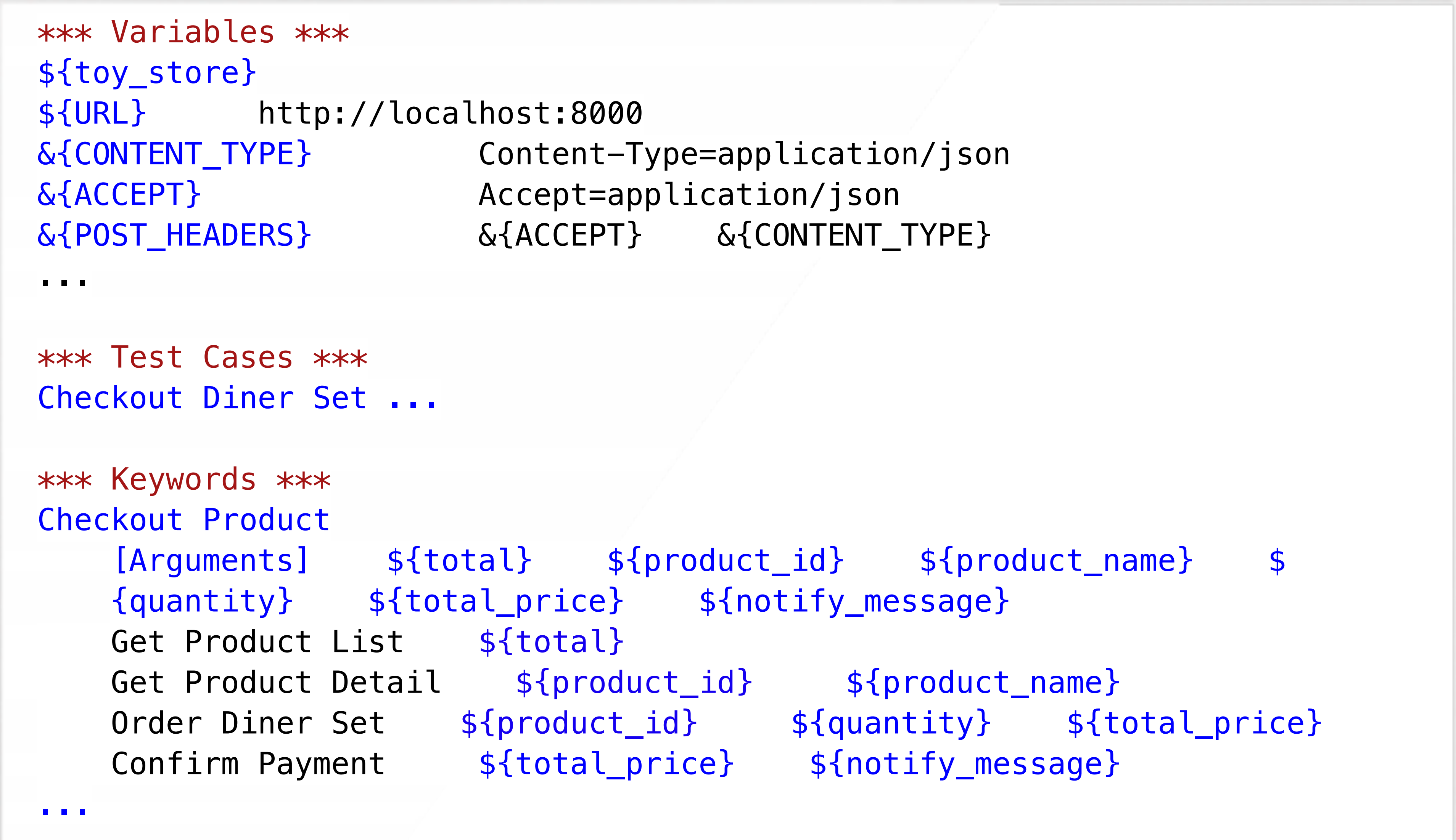
Run
Test Case Files



```
$ robot checkout-success-resources.robot
```

Test Cases
in Test Case Files

checkout-success-resources.robot



```
*** Variables ***
${toy_store}
${URL}      http://localhost:8000
&{CONTENT_TYPE}         Content-Type=application/json
&{ACCEPT}               Accept=application/json
&{POST_HEADERS}         &{ACCEPT}    &{CONTENT_TYPE}
...

*** Test Cases ***
Checkout Diner Set ...

*** Keywords ***
Checkout Product
    [Arguments]    ${total}    ${product_id}    ${product_name}    $
    {quantity}    ${total_price}    ${notify_message}
    Get Product List    ${total}
    Get Product Detail    ${product_id}     ${product_name}
    Order Diner Set    ${product_id}     ${quantity}    ${total_price}
    Confirm Payment    ${total_price}    ${notify_message}
...
```

Test Cases in Test Case Files

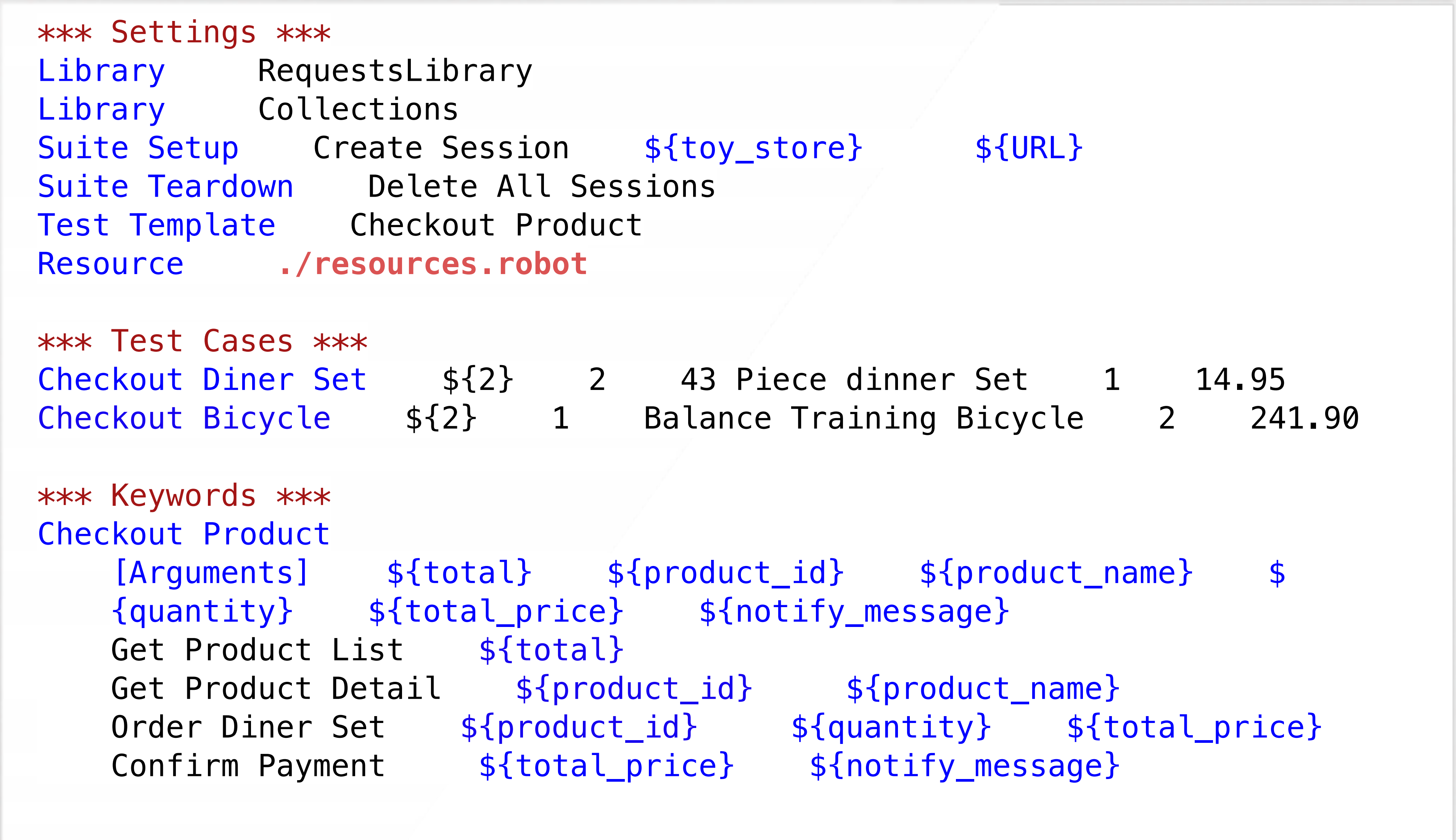
move

*** Variables ***

*** Keywords ***

to resources.robot

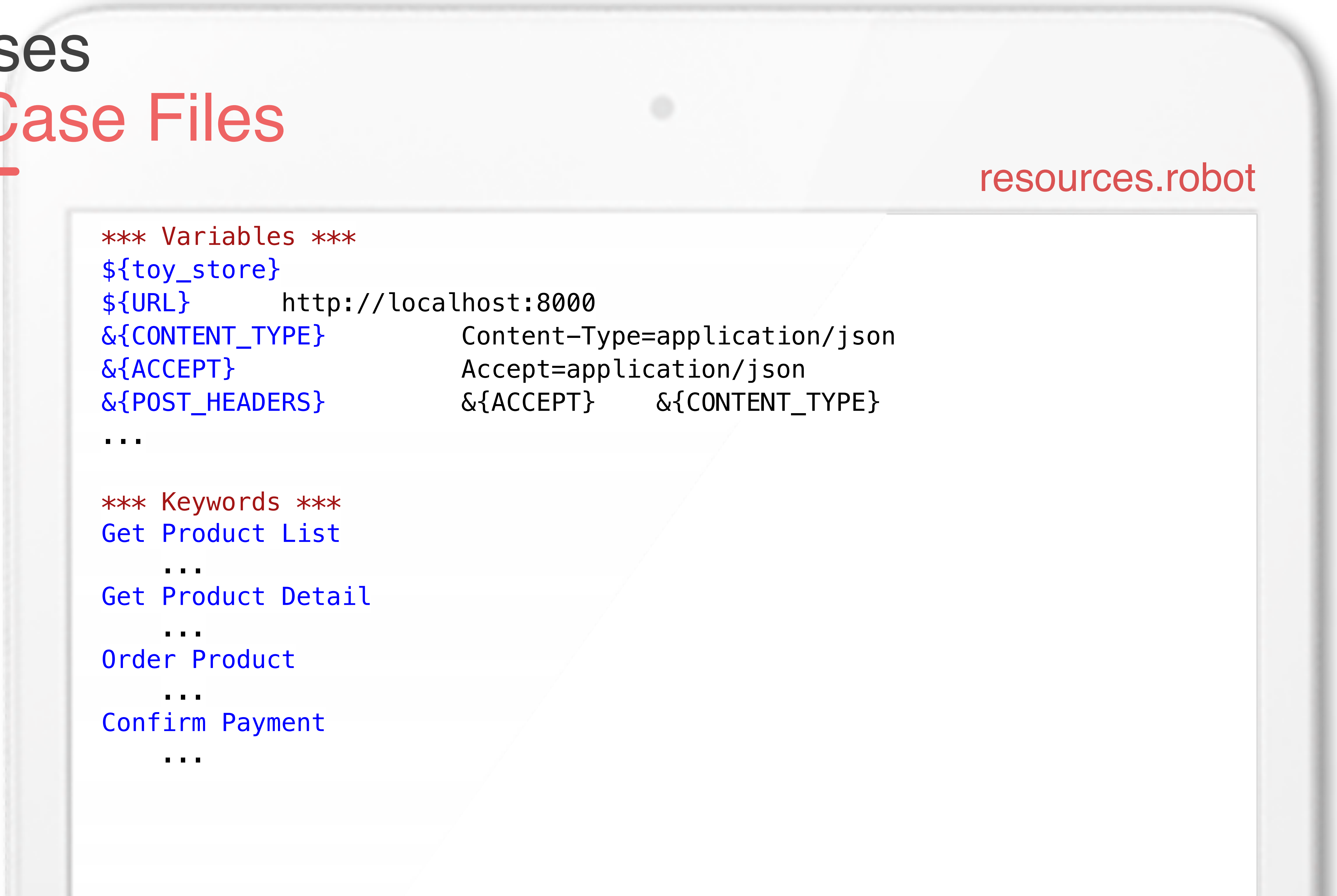
checkout-success-resources.robot



```
*** Settings ***
Library     RequestsLibrary
Library     Collections
Suite Setup    Create Session    ${toy_store}     ${URL}
Suite Teardown    Delete All Sessions
Test Template    Checkout Product
Resource     ./resources.robot

*** Test Cases ***
Checkout Diner Set    ${2}    2    43 Piece dinner Set    1    14.95
Checkout Bicycle    ${2}    1    Balance Training Bicycle    2    241.90

*** Keywords ***
Checkout Product
    [Arguments]    ${total}    ${product_id}    ${product_name}    $
    {quantity}    ${total_price}    ${notify_message}
    Get Product List    ${total}
    Get Product Detail    ${product_id}     ${product_name}
    Order Diner Set    ${product_id}     ${quantity}    ${total_price}
    Confirm Payment    ${total_price}    ${notify_message}
```

Test Cases in Test Case Files

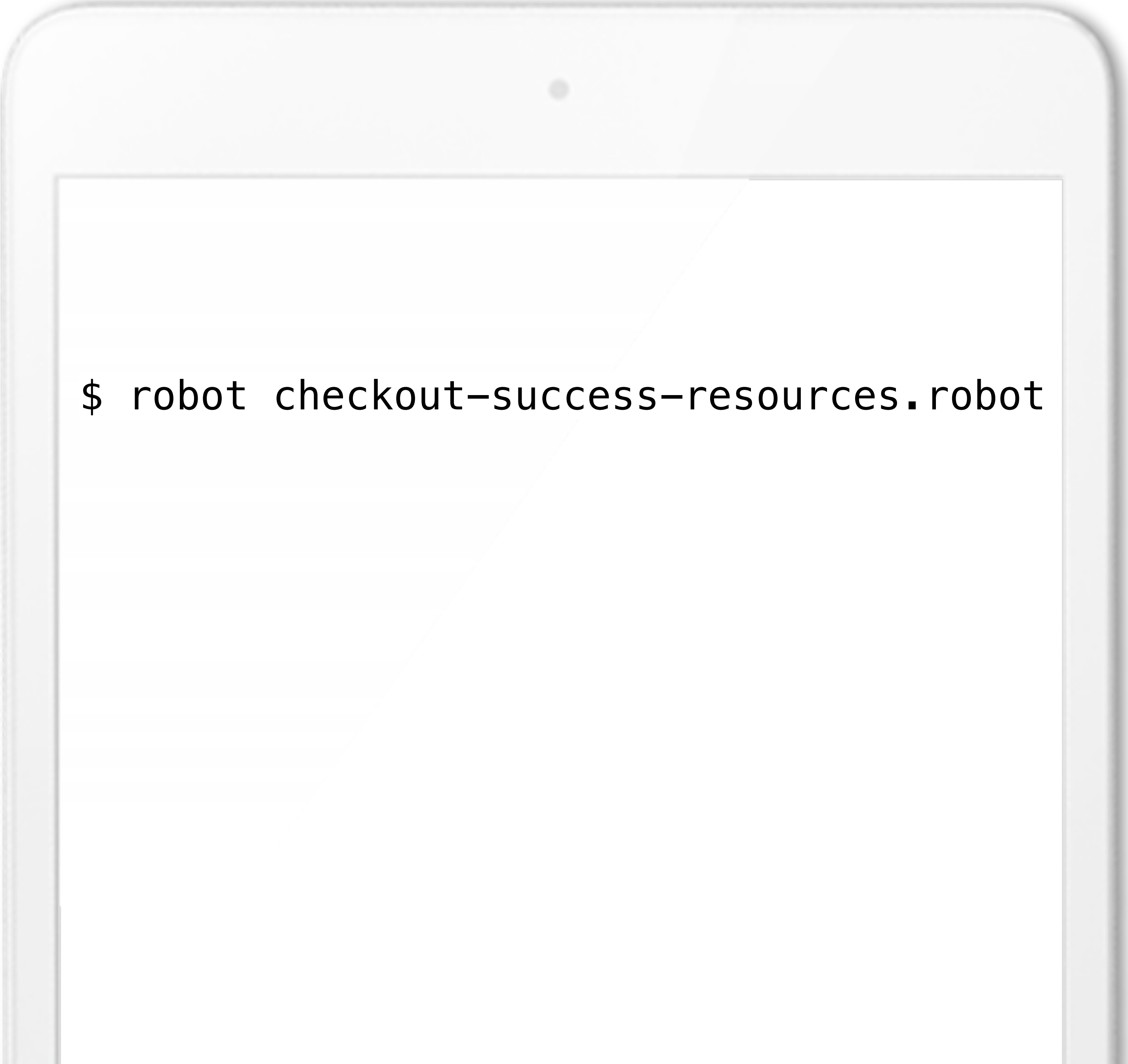
Hard Code!



resources.robot

```
*** Variables ***
${toy_store}
${URL}      http://localhost:8000
&{CONTENT_TYPE}         Content-Type=application/json
&{ACCEPT}               Accept=application/json
&{POST_HEADERS}         &{ACCEPT}    &{CONTENT_TYPE}
...

*** Keywords ***
Get Product List
    ...
Get Product Detail
    ...
Order Product
    ...
Confirm Payment
    ...
```

Run
Test Case Files



```
$ robot checkout-success-resources.robot
```

Robotframework Faker

lean more

© 2020 Siam Chamnankit Company Limited

Prerequisite(Global Install)

1. Python3

2. Robot Framework
   ```
   pip install robotframework
   ```

3. HTTP RequestsLibrary(Python)
   ```
   pip install Faker
   pip install robotframework-faker
   ```

4. Check the installed lib
   ```
   pip list
   ```

© 2020 Siam Chamnankit Company Limited

Robotframework Tag

lean more

© 2020 Siam Chamnankit Company Limited

Robotframework DateTime

lean more

© 2020 Siam Chamnankit Company Limited

Robotframework Data Driver

lean more

© 2020 Siam Chamnankit Company Limited

Prerequisite(Global Install)

1. Python3

2. Robot Framework
   `pip install robotframework`

3. HTTP RequestsLibrary(Python)
   `pip install robotframework-datadriver`

4. `Check the installed lib`
   `pip list`

© 2020 Siam Chamnankit Company Limited

thanks

30 slides