Space X Falcon 9 First Stage Landing Prediction

Web scraping Falcon 9 and Falcon Heavy Launches Records from Wikipedia

Estimated time needed: **40** minutes

In this lab, you will be performing web scraping to collect Falcon 9 historical launch records from a Wikipedia page titled `List of Falcon 9 and Falcon Heavy launches`

https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

Falcon 9 first stage will land successfully

Several examples of an unsuccessful landing are shown here:

More specifically, the launch records are stored in a HTML table shown below:

Objectives

Web scrap Falcon 9 launch records with `BeautifulSoup`:

- Extract a Falcon 9 launch records HTML table from Wikipedia
- Parse the table and convert it into a Pandas data frame

First let's import required packages for this lab

In [1]:
```
!pip3 install beautifulsoup4
!pip3 install requests
```

```
/opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages/secretstorage/dhcrypt
o.py:16: CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_
bytes instead
```

```
  from cryptography.utils import int_from_bytes
/opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages/secretstorage/util.py:
25: CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_bytes
instead
  from cryptography.utils import int_from_bytes
Requirement already satisfied: beautifulsoup4 in /opt/conda/envs/Python-3.7-OpenCE/l
ib/python3.7/site-packages (4.9.1)
Requirement already satisfied: soupsieve>1.2 in /opt/conda/envs/Python-3.7-OpenCE/li
b/python3.7/site-packages (from beautifulsoup4) (2.0.1)
/opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages/secretstorage/dhcrypt
o.py:16: CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_
bytes instead
  from cryptography.utils import int_from_bytes
/opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages/secretstorage/util.py:
25: CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_bytes
instead
  from cryptography.utils import int_from_bytes
Requirement already satisfied: requests in /opt/conda/envs/Python-3.7-OpenCE/lib/pyt
hon3.7/site-packages (2.25.1)
Requirement already satisfied: idna<3,>=2.5 in /opt/conda/envs/Python-3.7-OpenCE/li
b/python3.7/site-packages (from requests) (2.8)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /opt/conda/envs/Python-3.7-O
penCE/lib/python3.7/site-packages (from requests) (1.26.6)
Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/envs/Python-3.7-Open
CE/lib/python3.7/site-packages (from requests) (2021.5.30)
Requirement already satisfied: chardet<5,>=3.0.2 in /opt/conda/envs/Python-3.7-OpenC
E/lib/python3.7/site-packages (from requests) (3.0.4)
```

In [2]:

```
import sys

import requests
from bs4 import BeautifulSoup
import re
import unicodedata
import pandas as pd
```

and we will provide some helper functions for you to process web scraped HTML table

In [3]:

```
def date_time(table_cells):
    """
    This function returns the data and time from the HTML  table cell
    Input: the  element of a table data cell extracts extra row
    """
    return [data_time.strip() for data_time in list(table_cells.strings)][0:2]

def booster_version(table_cells):
    """
    This function returns the booster version from the HTML  table cell
    Input: the  element of a table data cell extracts extra row
    """
    out=''.join([booster_version for i,booster_version in enumerate( table_cells.str
    return out

def landing_status(table_cells):
    """
    This function returns the landing status from the HTML table cell
    Input: the  element of a table data cell extracts extra row
    """
```

```
    out=[i for i in table_cells.strings][0]
    return out


def get_mass(table_cells):
    mass=unicodedata.normalize("NFKD", table_cells.text).strip()
    if mass:
        mass.find("kg")
        new_mass=mass[0:mass.find("kg")+2]
    else:
        new_mass=0
    return new_mass


def extract_column_from_header(row):
    """
    This function returns the landing status from the HTML table cell
    Input: the  element of a table data cell extracts extra row
    """
    if (row.br):
        row.br.extract()
    if row.a:
        row.a.extract()
    if row.sup:
        row.sup.extract()

    colunm_name = ' '.join(row.contents)

    # Filter the digit and empty names
    if not(colunm_name.strip().isdigit()):
        colunm_name = colunm_name.strip()
        return colunm_name
```

To keep the lab tasks consistent, you will be asked to scrape the data from a snapshot of the `List of Falcon 9 and Falcon Heavy launches` Wikipage updated on `9th June 2021`

In [4]:
```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon
```

Next, request the HTML page from the above URL and get a `response` object

TASK 1: Request the Falcon9 Launch Wiki page from its URL

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

In [8]:
```
# use requests.get() method with the provided static_url
# assign the response to a object
html_data = requests.get(static_url).text
```

Create a `BeautifulSoup` object from the HTML `response`

In [9]:
```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
```

```
soup = BeautifulSoup(html_data, 'html5lib')
```

Print the page title to verify if the `BeautifulSoup` object was created properly

In [13]:
```
# Use soup.title attribute
soup.title
```

Out[13]:
```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

TASK 2: Extract all column/variable names from the HTML table header

Next, we want to collect all relevant column names from the HTML table header

Let's try to find all tables on the wiki page first. If you need to refresh your memory about `BeautifulSoup`, please check the external reference link towards the end of this lab

In [14]:
```
# Use the find_all function in the BeautifulSoup object, with element type `table`
# Assign the result to a list called `html_tables`
html_tables = soup.find_all('table')
```

Starting from the third table is our target table contains the actual launch records.

In [17]:
```
# Let's print the third table and check its content
first_launch_table = html_tables[2]
print(first_launch_table)
```

```
<table class="wikitable plainrowheaders collapsible" style="width: 100%;">
<tbody><tr>
<th scope="col">Flight No.
</th>
<th scope="col">Date and<br/>time (<a href="/wiki/Coordinated_Universal_Time" title
="Coordinated Universal Time">UTC</a>)
</th>
<th scope="col"><a href="/wiki/List_of_Falcon_9_first-stage_boosters" title="List of
Falcon 9 first-stage boosters">Version,<br/>Booster</a> <sup class="reference" id="c
ite_ref-booster_11-0"><a href="#cite_note-booster-11">[b]</a></sup>
</th>
<th scope="col">Launch site
</th>
<th scope="col">Payload<sup class="reference" id="cite_ref-Dragon_12-0"><a href="#ci
te_note-Dragon-12">[c]</a></sup>
</th>
<th scope="col">Payload mass
</th>
<th scope="col">Orbit
</th>
<th scope="col">Customer
</th>
<th scope="col">Launch<br/>outcome
</th>
<th scope="col"><a href="/wiki/Falcon_9_first-stage_landing_tests" title="Falcon 9 f
```

irst-stage landing tests">Booster
landing
</th></tr>
<tr>
<th rowspan="2" scope="row" style="text-align:center;">1
</th>
<td>4 June 2010,
18:45
</td>
<td>F9 v1.0<sup class="refer
ence" id="cite_ref-MuskMay2012_13-0">[7]</su
p>
B0003.1<sup class="reference" id="cite_ref-block_numbers_14-0"><a href="#cite
_note-block_numbers-14">[8]</sup>
</td>
<td><a href="/wiki/Cape_Canaveral_Space_Force_Station" title="Cape Canaveral Space F
orce Station">CCAFS,
<a href="/wiki/Cape_Canaveral_Space_Launch_Complex_40"
title="Cape Canaveral Space Launch Complex 40">SLC-40
</td>
<td><a href="/wiki/Dragon_Spacecraft_Qualification_Unit" title="Dragon Spacecraft Qu
alification Unit">Dragon Spacecraft Qualification Unit
</td>
<td>
</td>
<td>LEO
</td>
<td>SpaceX
</td>
<td class="table-success" style="background: LightGreen; color: black; vertical-alig
n: middle; text-align: center;">Success
</td>
<td class="table-failure" style="background: #ffbbbb; color: black; vertical-align:
middle; text-align: center;">Failure<sup class="reference" id="cite_ref-ns20110930_1
5-0">[9]</sup><sup class="reference" id="cite
_ref-16">[10]</sup>
<small>(parachute)</small>
</td></tr>
<tr>
<td colspan="9">First flight of Falcon 9 v1.0.<sup class="reference" id="cite_ref-sf
n20100604_17-0">[11]</sup> Used a boilerplat
e version of Dragon capsule which was not designed to separate from the second stag
e.<small>(more details below)</small> Attemp
ted to recover the first stage by parachuting it into the ocean, but it burned up on
reentry, before the parachutes even deployed.<sup class="reference" id="cite_ref-par
achute_18-0">[12]</sup>
</td></tr>
<tr>
<th rowspan="2" scope="row" style="text-align:center;">2
</th>
<td>8 December 2010,
15:43<sup class="reference" id="cite_ref-spaceflightnow_Cla
rk_Launch_Report_19-0">[1
3]</sup>
</td>
<td>F9 v1.0<sup class="refer
ence" id="cite_ref-MuskMay2012_13-1">[7]</su
p>
B0004.1<sup class="reference" id="cite_ref-block_numbers_14-1"><a href="#cite
_note-block_numbers-14">[8]</sup>
</td>
<td><a href="/wiki/Cape_Canaveral_Space_Force_Station" title="Cape Canaveral Space F
orce Station">CCAFS,
<a href="/wiki/Cape_Canaveral_Space_Launch_Complex_40"
title="Cape Canaveral Space Launch Complex 40">SLC-40
</td>
<td>Dragon <a class="mw-redi
rect" href="/wiki/COTS_Demo_Flight_1" title="COTS Demo Flight 1">demo flight C1<

```
br/>(Dragon C101)
</td>
<td>
</td>
<td><a href="/wiki/Low_Earth_orbit" title="Low Earth orbit">LEO</a> (<a href="/wiki/
International_Space_Station" title="International Space Station">ISS</a>)
</td>
<td><div class="plainlist">
<ul><li><a href="/wiki/NASA" title="NASA">NASA</a> (<a href="/wiki/Commercial_Orbita
l_Transportation_Services" title="Commercial Orbital Transportation Services">COTS</
a>)</li>
<li><a href="/wiki/National_Reconnaissance_Office" title="National Reconnaissance Of
fice">NRO</a></li></ul>
</div>
</td>
<td class="table-success" style="background: LightGreen; color: black; vertical-alig
n: middle; text-align: center;">Success<sup class="reference" id="cite_ref-ns2011093
0_15-1"><a href="#cite_note-ns20110930-15">[9]</a></sup>
</td>
<td class="table-failure" style="background: #ffbbbb; color: black; vertical-align:
middle; text-align: center;">Failure<sup class="reference" id="cite_ref-ns20110930_1
5-2"><a href="#cite_note-ns20110930-15">[9]</a></sup><sup class="reference" id="cite
_ref-20"><a href="#cite_note-20">[14]</a></sup><br/><small>(parachute)</small>
</td></tr>
<tr>
<td colspan="9">Maiden flight of <a class="mw-redirect" href="/wiki/Dragon_capsule"
title="Dragon capsule">Dragon capsule</a>, consisting of over 3 hours of testing thr
uster maneuvering and reentry.<sup class="reference" id="cite_ref-spaceflightnow_Cla
rk_unleashing_Dragon_21-0"><a href="#cite_note-spaceflightnow_Clark_unleashing_Drago
n-21">[15]</a></sup> Attempted to recover the first stage by parachuting it into the
ocean, but it disintegrated upon reentry, before the parachutes were deployed.<sup c
lass="reference" id="cite_ref-parachute_18-1"><a href="#cite_note-parachute-18">[12]
</a></sup> <small>(<a href="#COTS_demo_missions">more details below</a>)</small> It
also included two <a href="/wiki/CubeSat" title="CubeSat">CubeSats</a>,<sup class="r
eference" id="cite_ref-NRO_Taps_Boeing_for_Next_Batch_of_CubeSats_22-0"><a href="#ci
te_note-NRO_Taps_Boeing_for_Next_Batch_of_CubeSats-22">[16]</a></sup> and a wheel of
<a href="/wiki/Brou%C3%A8re" title="Brouère">Brouère</a> cheese.
</td></tr>
<tr>
<th rowspan="2" scope="row" style="text-align:center;">3
</th>
<td>22 May 2012,<br/>07:44<sup class="reference" id="cite_ref-BBC_new_era_23-0"><a h
ref="#cite_note-BBC_new_era-23">[17]</a></sup>
</td>
<td><a href="/wiki/Falcon_9_v1.0" title="Falcon 9 v1.0">F9 v1.0</a><sup class="refer
ence" id="cite_ref-MuskMay2012_13-2"><a href="#cite_note-MuskMay2012-13">[7]</a></su
p><br/>B0005.1<sup class="reference" id="cite_ref-block_numbers_14-2"><a href="#cite
_note-block_numbers-14">[8]</a></sup>
</td>
<td><a href="/wiki/Cape_Canaveral_Space_Force_Station" title="Cape Canaveral Space F
orce Station">CCAFS</a>,<br/><a href="/wiki/Cape_Canaveral_Space_Launch_Complex_40"
title="Cape Canaveral Space Launch Complex 40">SLC-40</a>
</td>
<td><a href="/wiki/SpaceX_Dragon" title="SpaceX Dragon">Dragon</a> <a class="mw-redi
rect" href="/wiki/Dragon_C2%2B" title="Dragon C2+">demo flight C2+</a><sup class="re
ference" id="cite_ref-C2_24-0"><a href="#cite_note-C2-24">[18]</a></sup><br/>(Dragon
C102)
</td>
<td>525 kg (1,157 lb)<sup class="reference" id="cite_ref-25"><a href="#cite_note-2
5">[19]</a></sup>
```

```
</td>
<td><a href="/wiki/Low_Earth_orbit" title="Low Earth orbit">LEO</a> (<a href="/wiki/
International_Space_Station" title="International Space Station">ISS</a>)
</td>
<td><a href="/wiki/NASA" title="NASA">NASA</a> (<a href="/wiki/Commercial_Orbital_Tr
ansportation_Services" title="Commercial Orbital Transportation Services">COTS</a>)
</td>
<td class="table-success" style="background: LightGreen; color: black; vertical-alig
n: middle; text-align: center;">Success<sup class="reference" id="cite_ref-26"><a hr
ef="#cite_note-26">[20]</a></sup>
</td>
<td class="table-noAttempt" style="background: #ececec; color: black; vertical-alig
n: middle; white-space: nowrap; text-align: center;">No attempt
</td></tr>
<tr>
<td colspan="9">Dragon spacecraft demonstrated a series of tests before it was allow
ed to approach the <a href="/wiki/International_Space_Station" title="International
Space Station">International Space Station</a>. Two days later, it became the first
commercial spacecraft to board the ISS.<sup class="reference" id="cite_ref-BBC_new_e
ra_23-1"><a href="#cite_note-BBC_new_era-23">[17]</a></sup> <small>(<a href="#COTS_d
emo_missions">more details below</a>)</small>
</td></tr>
<tr>
<th rowspan="3" scope="row" style="text-align:center;">4
</th>
<td rowspan="2">8 October 2012,<br/>00:35<sup class="reference" id="cite_ref-SFN_LLo
g_27-0"><a href="#cite_note-SFN_LLog-27">[21]</a></sup>
</td>
<td rowspan="2"><a href="/wiki/Falcon_9_v1.0" title="Falcon 9 v1.0">F9 v1.0</a><sup
class="reference" id="cite_ref-MuskMay2012_13-3"><a href="#cite_note-MuskMay2012-1
3">[7]</a></sup><br/>B0006.1<sup class="reference" id="cite_ref-block_numbers_14-3">
<a href="#cite_note-block_numbers-14">[8]</a></sup>
</td>
<td rowspan="2"><a href="/wiki/Cape_Canaveral_Space_Force_Station" title="Cape Canav
eral Space Force Station">CCAFS</a>,<br/><a href="/wiki/Cape_Canaveral_Space_Launch_
Complex_40" title="Cape Canaveral Space Launch Complex 40">SLC-40</a>
</td>
<td><a href="/wiki/SpaceX_CRS-1" title="SpaceX CRS-1">SpaceX CRS-1</a><sup class="re
ference" id="cite_ref-sxManifest20120925_28-0"><a href="#cite_note-sxManifest2012092
5-28">[22]</a></sup><br/>(Dragon C103)
</td>
<td>4,700 kg (10,400 lb)
</td>
<td><a href="/wiki/Low_Earth_orbit" title="Low Earth orbit">LEO</a> (<a href="/wiki/
International_Space_Station" title="International Space Station">ISS</a>)
</td>
<td><a href="/wiki/NASA" title="NASA">NASA</a> (<a href="/wiki/Commercial_Resupply_S
ervices" title="Commercial Resupply Services">CRS</a>)
</td>
<td class="table-success" style="background: LightGreen; color: black; vertical-alig
n: middle; text-align: center;">Success
</td>
<td rowspan="2" style="background:#ececec; text-align:center;"><span class="nowrap">
No attempt</span>
</td></tr>
<tr>
<td><a href="/wiki/Orbcomm_(satellite)" title="Orbcomm (satellite)">Orbcomm-OG2</a><
sup class="reference" id="cite_ref-Orbcomm_29-0"><a href="#cite_note-Orbcomm-29">[2
3]</a></sup>
</td>
```

<td>172 kg (379 lb)^{[24]}
</td>
<td>LEO
</td>
<td>Orbcomm
</td>
<td class="table-partial" style="background: wheat; color: black; vertical-align: middle; text-align: center;">Partial failure^{[25]}
</td></tr>
<tr>
<td colspan="9">CRS-1 was successful, but the secondary payload was inserted into an abnormally low orbit and subsequently lost. This was due to one of the nine Merlin engines shutting down during the launch, and NASA declining a second reignition, as per ISS visiting vehicle safety rules, the primary payload owner is contractually allowed to decline a second reignition. NASA stated that this was because SpaceX could not guarantee a high enough likelihood of the second stage completing the second burn successfully which was required to avoid any risk of secondary payload's collision with the ISS.^{[26]}^{[27]}^{[28]}
</td></tr>
<tr>
<th rowspan="2" scope="row" style="text-align:center;">5
</th>
<td>1 March 2013,
15:10
</td>
<td>F9 v1.0^{[7]}
B0007.1^{[8]}
</td>
<td>CCAFS,
SLC-40
</td>
<td>SpaceX CRS-2^{[22]}
(Dragon C104)
</td>
<td>4,877 kg (10,752 lb)
</td>
<td>LEO (ISS)
</td>
<td>NASA (CRS)
</td>
<td class="table-success" style="background: LightGreen; color: black; vertical-align: middle; text-align: center;">Success
</td>
<td class="table-noAttempt" style="background: #ececec; color: black; vertical-align: middle; white-space: nowrap; text-align: center;">No attempt
</td></tr>
<tr>

<td colspan="9">Last launch of the original Falcon 9 v1.0 launch vehicle, first use of the unpressurized trunk section of Dragon.^{[29]}
</td></tr>
<tr>
<th rowspan="2" scope="row" style="text-align:center;">6
</th>
<td>29 September 2013,
16:00^{[30]}
</td>
<td>F9 v1.1^{[7]}
B1003^{[8]}
</td>
<td>VAFB,
SLC-4E
</td>
<td>CASSIOPE^{[22]}^{[31]}
</td>
<td>500 kg (1,100 lb)
</td>
<td>Polar orbit LEO
</td>
<td>MDA
</td>
<td class="table-success" style="background: LightGreen; color: black; vertical-align: middle; text-align: center;">Success^{[30]}
</td>
<td class="table-no2" style="background: #ffdddd; color: black; vertical-align: middle; text-align: center;">Uncontrolled
<small>(ocean)</small>^{[d]}
</td></tr>
<tr>
<td colspan="9">First commercial mission with a private customer, first launch from Vandenberg, and demonstration flight of Falcon 9 v1.1 with an improved 13-tonne to LEO capacity.^{[29]} After separation from the second stage carrying Canadian commercial and scientific satellites, the first stage booster performed a controlled reentry,^{[32]} and an ocean touchdown test for the first time. This provided good test data, even though the booster started rolling as it neared the ocean, leading to the shutdown of the central engine as the roll depleted it of fuel, resulting in a hard impact with the ocean.^{[30]} This was the first known attempt of a rocket engine being lit to perform a supersonic retro propulsion, and allowed SpaceX to enter a public-private partnership with NASA and its Mars entry, descent, and landing technologies research projects.^{[33]} <small>(more details below)</small>
</td></tr>

```
<tr>
<th rowspan="2" scope="row" style="text-align:center;">7
</th>
<td>3 December 2013,<br/>22:41<sup class="reference" id="cite_ref-sfn_wwls20130624_4
1-0"><a href="#cite_note-sfn_wwls20130624-41">[34]</a></sup>
</td>
<td><a href="/wiki/Falcon_9_v1.1" title="Falcon 9 v1.1">F9 v1.1</a><br/>B1004
</td>
<td><a href="/wiki/Cape_Canaveral_Space_Force_Station" title="Cape Canaveral Space F
orce Station">CCAFS</a>,<br/><a href="/wiki/Cape_Canaveral_Space_Launch_Complex_40"
title="Cape Canaveral Space Launch Complex 40">SLC-40</a>
</td>
<td><a href="/wiki/SES-8" title="SES-8">SES-8</a><sup class="reference" id="cite_ref
-sxManifest20120925_28-3"><a href="#cite_note-sxManifest20120925-28">[22]</a></sup><
sup class="reference" id="cite_ref-spx-pr_42-0"><a href="#cite_note-spx-pr-42">[35]
</a></sup><sup class="reference" id="cite_ref-aw20110323_43-0"><a href="#cite_note-a
w20110323-43">[36]</a></sup>
</td>
<td>3,170 kg (6,990 lb)
</td>
<td><a href="/wiki/Geostationary_transfer_orbit" title="Geostationary transfer orbi
t">GTO</a>
</td>
<td><a href="/wiki/SES_S.A." title="SES S.A.">SES</a>
</td>
<td class="table-success" style="background: LightGreen; color: black; vertical-alig
n: middle; text-align: center;">Success<sup class="reference" id="cite_ref-SNMission
Status7_44-0"><a href="#cite_note-SNMissionStatus7-44">[37]</a></sup>
</td>
<td class="table-noAttempt" style="background: #ececec; color: black; vertical-alig
n: middle; white-space: nowrap; text-align: center;">No attempt<br/><sup class="refe
rence" id="cite_ref-sf10120131203_45-0"><a href="#cite_note-sf10120131203-45">[38]</
a></sup>
</td></tr>
<tr>
<td colspan="9">First <a href="/wiki/Geostationary_transfer_orbit" title="Geostation
ary transfer orbit">Geostationary transfer orbit</a> (GTO) launch for Falcon 9,<sup
class="reference" id="cite_ref-spx-pr_42-1"><a href="#cite_note-spx-pr-42">[35]</a>
</sup> and first successful reignition of the second stage.<sup class="reference" id
="cite_ref-46"><a href="#cite_note-46">[39]</a></sup> SES-8 was inserted into a <a h
ref="/wiki/Geostationary_transfer_orbit" title="Geostationary transfer orbit">Super-
Synchronous Transfer Orbit</a> of 79,341 km (49,300 mi) in apogee with an <a href="/
wiki/Orbital_inclination" title="Orbital inclination">inclination</a> of 20.55° to t
he <a href="/wiki/Equator" title="Equator">equator</a>.
</td></tr></tbody></table>
```

You should able to see the columns names embedded in the table header elements `<th>` as follows:

```
<tr>
<th scope="col">Flight No.
</th>
<th scope="col">Date and<br/>time (<a
href="/wiki/Coordinated_Universal_Time" title="Coordinated Universal
Time">UTC</a>)
</th>
<th scope="col"><a href="/wiki/List_of_Falcon_9_first-
```

```
stage_boosters" title="List of Falcon 9 first-stage
boosters">Version,<br/>Booster</a> <sup class="reference"
id="cite_ref-booster_11-0"><a href="#cite_note-booster-11">[b]</a>
</sup>
</th>
<th scope="col">Launch site
</th>
<th scope="col">Payload<sup class="reference" id="cite_ref-
Dragon_12-0"><a href="#cite_note-Dragon-12">[c]</a></sup>
</th>
<th scope="col">Payload mass
</th>
<th scope="col">Orbit
</th>
<th scope="col">Customer
</th>
<th scope="col">Launch<br/>outcome
</th>
<th scope="col"><a href="/wiki/Falcon_9_first-stage_landing_tests"
title="Falcon 9 first-stage landing tests">Booster<br/>landing</a>
</th></tr>
```

Next, we just need to iterate through the `<th>` elements and apply the provided `extract_column_from_header()` to extract column name one by one

In [21]:
```
column_names = []

# Apply find_all() function with `th` element on first_launch_table
# Iterate each th element and apply the provided extract_column_from_header() to get
# Append the Non-empty column name (`if name is not None and len(name) > 0`) into a
for row in first_launch_table.find_all('th'):
    name = extract_column_from_header(row)
    if(name != None and len(name) > 0):
        column_names.append(name)
```

Check the extracted column names

In [37]:
```
print(column_names)
```

```
['Flight No.', 'Date and time ( )', 'Launch site', 'Payload', 'Payload mass', 'Orbi
t', 'Customer', 'Launch outcome']
```

TASK 3: Create a data frame by parsing the launch HTML tables

We will create an empty dictionary with keys from the extracted column names in the previous task. Later, this dictionary will be converted into a Pandas dataframe

In [38]:
```
launch_dict= dict.fromkeys(column_names)
```

```
# Remove an irrelvant column
del launch_dict['Date and time ( )']

# Let's initial the launch_dict with each value to be an empty list
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

Next, we just need to fill up the `launch_dict` with launch records extracted from table rows.

Usually, HTML tables in Wiki pages are likely to contain unexpected annotations and other types of noises, such as reference links `B0004.1[8]`, missing values `N/A [e]`, inconsistent formatting, etc.

To simplify the parsing process, we have provided an incomplete code snippet below to help you to fill up the `launch_dict`. Please complete the following code snippet with TODOs or you can choose to write your own logic to parse all launch tables:

In [39]:

```
extracted_row = 0
#Extract each table
for table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders
   # get table row
    for rows in table.find_all("tr"):
        #check to see if first table heading is as number corresponding to launch a
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
        else:
            flag=False
        #get table element
        row=rows.find_all('td')
        #if it is number save cells in a dictonary
        if flag:
            extracted_row += 1
            # Flight Number value
            # TODO: Append the flight_number into launch_dict with key `Flight No.`
            launch_dict['Flight No.'].append(flight_number)
            print(flight_number)
            datatimelist=date_time(row[0])

            # Date value
            # TODO: Append the date into launch_dict with key `Date`
            date = datatimelist[0].strip(',')
            print(date)
            launch_dict['Date'].append(date)
```

```
            # Time value
            # TODO: Append the time into launch_dict with key `Time`
            time = datatimelist[1]
            print(time)
            launch_dict['Time'].append(time)

            # Booster version
            # TODO: Append the bv into launch_dict with key `Version Booster`
            bv=booster_version(row[1])
            if not(bv):
                bv=row[1].a.string
            print(bv)
            launch_dict['Version Booster'].append(bv)

            # Launch Site
            # TODO: Append the bv into launch_dict with key `Launch Site`
            launch_site = row[2].a.string
            print(launch_site)
            launch_dict['Launch site'].append(launch_site)

            # Payload
            # TODO: Append the payload into launch_dict with key `Payload`
            payload = row[3].a.string
            print(payload)
            launch_dict['Payload'].append(payload)

            # Payload Mass
            # TODO: Append the payload_mass into launch_dict with key `Payload mass`
            payload_mass = get_mass(row[4])
            print(payload)
            launch_dict['Payload mass'].append(payload_mass)

            # Orbit
            # TODO: Append the orbit into launch_dict with key `Orbit`
            orbit = row[5].a.string
            print(orbit)
            launch_dict['Orbit'].append(orbit)

            # Customer
            # TODO: Append the customer into launch_dict with key `Customer`
            customer = row[6].text.strip()
            print(customer)
            launch_dict['Customer'].append(customer)

            # Launch outcome
            # TODO: Append the launch_outcome into launch_dict with key `Launch outc
            launch_outcome = list(row[7].strings)[0]
            print(launch_outcome)
            launch_dict['Launch outcome'].append(launch_outcome)

            # Booster landing
            # TODO: Append the launch_outcome into launch_dict with key `Booster lan
            booster_landing = landing_status(row[8])
            print(booster_landing)
            launch_dict['Booster landing'].append(booster_landing)
```

```
1
4 June 2010
```

18:45
F9 v1.0B0003.1
CCAFS
Dragon Spacecraft Qualification Unit
Dragon Spacecraft Qualification Unit
LEO
SpaceX
Success

Failure
2
8 December 2010
15:43
F9 v1.0B0004.1
CCAFS
Dragon
Dragon
LEO
NASA (COTS)
NRO
Success
Failure
3
22 May 2012
07:44
F9 v1.0B0005.1
CCAFS
Dragon
Dragon
LEO
NASA (COTS)
Success
No attempt

4
8 October 2012
00:35
F9 v1.0B0006.1
CCAFS
SpaceX CRS-1
SpaceX CRS-1
LEO
NASA (CRS)
Success

No attempt
5
1 March 2013
15:10
F9 v1.0B0007.1
CCAFS
SpaceX CRS-2
SpaceX CRS-2
LEO
NASA (CRS)
Success

No attempt

6

29 September 2013
16:00
F9 v1.1B1003
VAFB
CASSIOPE
CASSIOPE
Polar orbit
MDA
Success
Uncontrolled
7
3 December 2013
22:41
F9 v1.1
CCAFS
SES-8
SES-8
GTO
SES
Success
No attempt
8
6 January 2014
22:06
F9 v1.1
CCAFS
Thaicom 6
Thaicom 6
GTO
Thaicom
Success
No attempt
9
18 April 2014
19:25
F9 v1.1
Cape Canaveral
SpaceX CRS-3
SpaceX CRS-3
LEO
NASA (CRS)
Success

Controlled
10
14 July 2014
15:15
F9 v1.1
Cape Canaveral
Orbcomm-OG2
Orbcomm-OG2
LEO
Orbcomm
Success
Controlled
11
5 August 2014
08:00
F9 v1.1
Cape Canaveral

AsiaSat 8
AsiaSat 8
GTO
AsiaSat
Success
No attempt
12
7 September 2014
05:00
F9 v1.1
Cape Canaveral
AsiaSat 6
AsiaSat 6
GTO
AsiaSat
Success
No attempt

13
21 September 2014
05:52
F9 v1.1
Cape Canaveral
SpaceX CRS-4
SpaceX CRS-4
LEO
NASA (CRS)
Success
Uncontrolled
14
10 January 2015
09:47
F9 v1.1
Cape Canaveral
SpaceX CRS-5
SpaceX CRS-5
LEO
NASA (CRS)
Success
Failure
15
11 February 2015
23:03
F9 v1.1
Cape Canaveral
DSCOVR
DSCOVR
HEO
USAF
NASA
NOAA
Success

Controlled
16
2 March 2015
03:50
F9 v1.1
Cape Canaveral
ABS-3A

ABS-3A
GTO
ABS
Eutelsat
Success

No attempt
17
14 April 2015
20:10
F9 v1.1
Cape Canaveral
SpaceX CRS-6
SpaceX CRS-6
LEO
NASA (CRS)
Success

Failure
18
27 April 2015
23:03
F9 v1.1
Cape Canaveral
TürkmenÄlem 52°E / MonacoSAT
TürkmenÄlem 52°E / MonacoSAT
GTO
Turkmenistan NationalSpace Agency[88]
Success

No attempt
19
28 June 2015
14:21
F9 v1.1
Cape Canaveral
SpaceX CRS-7
SpaceX CRS-7
LEO
NASA (CRS)
Failure
Precluded
20
22 December 2015
01:29
F9 FT
Cape Canaveral
Orbcomm-OG2
Orbcomm-OG2
LEO
Orbcomm
Success

Success
21
17 January 2016
18:42
F9 v1.1
VAFB
Jason-3

Jason-3
LEO
NASA (LSP)
NOAA
CNES
Success

Failure
22
4 March 2016
23:35
F9 FT
Cape Canaveral
SES-9
SES-9
GTO
SES
Success

Failure
23
8 April 2016
20:43
F9 FT
Cape Canaveral
SpaceX CRS-8
SpaceX CRS-8
LEO
NASA (CRS)
Success
Success
24
6 May 2016
05:21
F9 FT
Cape Canaveral
JCSAT-14
JCSAT-14
GTO
SKY Perfect JSAT Group
Success

Success
25
27 May 2016
21:39
F9 FT
Cape Canaveral
Thaicom 8
Thaicom 8
GTO
Thaicom
Success

Success
26
15 June 2016
14:29
F9 FT
Cape Canaveral

ABS-2A
ABS-2A
GTO
ABS
Eutelsat
Success

Failure
27
18 July 2016
04:45
F9 FT
Cape Canaveral
SpaceX CRS-9
SpaceX CRS-9
LEO
NASA (CRS)
Success

Success
28
14 August 2016
05:26
F9 FT
Cape Canaveral
JCSAT-16
JCSAT-16
GTO
SKY Perfect JSAT Group
Success

Success
29
14 January 2017
17:54
F9 FT
VAFB
Iridium NEXT
Iridium NEXT
Polar
Iridium Communications
Success

Success
30
19 February 2017
14:39
F9 FT
KSC
SpaceX CRS-10
SpaceX CRS-10
LEO
NASA (CRS)
Success

Success
31
16 March 2017
06:00
F9 FT

KSC
EchoStar 23
EchoStar 23
GTO
EchoStar
Success

No attempt
32
30 March 2017
22:27
F9 FT♺
KSC
SES-10
SES-10
GTO
SES
Success
Success
33
1 May 2017
11:15
F9 FT
KSC
NROL-76
NROL-76
LEO
NRO
Success

Success
34
15 May 2017
23:21
F9 FT
KSC
Inmarsat-5 F4
Inmarsat-5 F4
GTO
Inmarsat
Success

No attempt
35
3 June 2017
21:07
F9 FT
KSC
SpaceX CRS-11
SpaceX CRS-11
LEO
NASA (CRS)
Success

Success
36
23 June 2017
19:10
F9 FTB1029.2
KSC

BulgariaSat-1
BulgariaSat-1
GTO
Bulsatcom
Success

Success
37
25 June 2017
20:25
F9 FT
VAFB
Iridium NEXT
Iridium NEXT
LEO
Iridium Communications
Success

Success
38
5 July 2017
23:38
F9 FT
KSC
Intelsat 35e
Intelsat 35e
GTO
Intelsat
Success

No attempt
39
14 August 2017
16:31
F9 B4
KSC
SpaceX CRS-12
SpaceX CRS-12
LEO
NASA (CRS)
Success

Success
40
24 August 2017
18:51
F9 FT
VAFB
Formosat-5
Formosat-5
SSO
NSPO
Success

Success
41
7 September 2017
14:00
F9 B4
KSC

Boeing X-37B
Boeing X-37B
LEO
USAF
Success

Success
42
9 October 2017
12:37
F9 B4
VAFB
Iridium NEXT
Iridium NEXT
Polar
Iridium Communications
Success

Success
43
11 October 2017
22:53:00
F9 FTB1031.2
KSC
SES-11
SES-11
GTO
SES S.A.
EchoStar
Success

Success
44
30 October 2017
19:34
F9 B4
KSC
Koreasat 5A
Koreasat 5A
GTO
KT Corporation
Success

Success
45
15 December 2017
15:36
F9 FTB1035.2
Cape Canaveral
SpaceX CRS-13
SpaceX CRS-13
LEO
NASA (CRS)
Success

Success
46
23 December 2017
01:27
F9 FTB1036.2

VAFB
Iridium NEXT
Iridium NEXT
Polar
Iridium Communications
Success
Controlled
47
8 January 2018
01:00
F9 B4
CCAFS
Zuma
Zuma
LEO
Northrop Grumman [f][238]
Success
Success
48
31 January 2018
21:25
F9 FTB1032.2
CCAFS
GovSat-1
GovSat-1
GTO
SES
Success
Controlled
49
22 February 2018
14:17
F9 FTB1038.2
VAFB
Paz
Paz
SSO
Hisdesat
exactEarth
SpaceX
Success
No attempt
50
6 March 2018
05:33
F9 B4
CCAFS
Hispasat 30W-6
Hispasat 30W-6
GTO
Hispasat[277]
NovaWurks
Success
No attempt
51
30 March 2018
14:14
F9 B4B1041.2
VAFB
Iridium NEXT

Iridium NEXT
Polar
Iridium Communications
Success
No attempt
52
2 April 2018
20:30
F9 B4B1039.2
CCAFS
SpaceX CRS-14
SpaceX CRS-14
LEO
NASA (CRS)
Success
No attempt
53
18 April 2018
22:51
F9 B4
CCAFS
Transiting Exoplanet Survey Satellite
Transiting Exoplanet Survey Satellite
HEO
NASA (LSP)
Success
Success
54
11 May 2018
20:14
F9 B5B1046.1
KSC
Bangabandhu-1
Bangabandhu-1
GTO
Thales-Alenia / BTRC
Success
Success
55
22 May 2018
19:47
F9 B4B1043.2
VAFB
Iridium NEXT
Iridium NEXT
Polar
Iridium Communications
GFZ • NASA
Success
No attempt
56
4 June 2018
04:45
F9 B4B1040.2
CCAFS
SES-12
SES-12
GTO
SES
Success

No attempt
57
29 June 2018
09:42
F9 B4B1045.2
CCAFS
SpaceX CRS-15
SpaceX CRS-15
LEO
NASA (CRS)
Success
No attempt
58
22 July 2018
05:50
F9 B5
CCAFS
Telstar 19V
Telstar 19V
GTO
Telesat
Success
Success
59
25 July 2018
11:39
F9 B5B1048
VAFB
Iridium NEXT
Iridium NEXT
Polar
Iridium Communications
Success
Success
60
7 August 2018
05:18
F9 B5B1046.2
CCAFS
Merah Putih
Merah Putih
GTO
Telkom Indonesia
Success
Success
61
10 September 2018
04:45
F9 B5
CCAFS
Telstar 18V
Telstar 18V
GTO
Telesat
Success
Success
62
8 October 2018
02:22
F9 B5B1048.2

VAFB
SAOCOM 1A
SAOCOM 1A
SSO
CONAE
Success
Success
63
15 November 2018
20:46
F9 B5B1047.2
KSC
Es'hail 2
Es'hail 2
GTO
Es'hailSat
Success
Success
64
3 December 2018
18:34:05
F9 B5B1046.3
VAFB
SSO-A
SSO-A
SSO
Spaceflight Industries
Success
Success
65
5 December 2018
18:16
F9 B5
CCAFS
SpaceX CRS-16
SpaceX CRS-16
LEO
NASA (CRS)
Success

Failure
66
23 December 2018
13:51
F9 B5
CCAFS
GPS III
GPS III
MEO
USAF
Success
No attempt
67
11 January 2019
15:31
F9 B5B1049.2
VAFB
Iridium NEXT
Iridium NEXT
Polar

Iridium Communications
Success

Success
68
22 February 2019
01:45
F9 B5B1048.3
CCAFS
Nusantara Satu
Nusantara Satu
GTO
PSN
SpaceIL / IAI
Air Force Research
Success

Success
69
2 March 2019
07:49
F9 B5[268]
KSC
Crew Dragon Demo-1
Crew Dragon Demo-1
LEO
NASA (CCD)
Success

Success
70
4 May 2019
06:48
F9 B5
CCAFS
SpaceX CRS-17
SpaceX CRS-17
LEO
NASA (CRS)
Success

Success
71
24 May 2019
02:30
F9 B5B1049.3
CCAFS
Starlink
Starlink
LEO
SpaceX
Success

Success
72
12 June 2019
14:17
F9 B5B1051.2
VAFB
RADARSAT Constellation

RADARSAT Constellation
SSO
Canadian Space Agency (CSA)
Success

Success
73
25 July 2019
22:01
F9 B5B1056.2
CCAFS
SpaceX CRS-18
SpaceX CRS-18
LEO
NASA (CRS)
Success

Success
74
6 August 2019
23:23
F9 B5B1047.3
CCAFS
AMOS-17
AMOS-17
GTO
Spacecom
Success

No attempt
75
11 November 2019
14:56
F9 B5
CCAFS
Starlink
Starlink
LEO
SpaceX
Success

Success
76
5 December 2019
17:29
F9 B5
CCAFS
SpaceX CRS-19
SpaceX CRS-19
LEO
NASA (CRS)
Success

Success
77
17 December 2019
00:10
F9 B5B1056.3
CCAFS
JCSat-18

JCSat-18
GTO
Sky Perfect JSATKacific 1
Success

Success
78
7 January 2020
02:19:21
F9 B5
CCAFS
Starlink
Starlink
LEO
SpaceX
Success

Success
79
19 January 2020
15:30
F9 B5
KSC
Crew Dragon in-flight abort test
Crew Dragon in-flight abort test
Sub-orbital
NASA (CTS)[497]
Success

No attempt

80
29 January 2020
14:07
F9 B5
CCAFS
Starlink
Starlink
LEO
SpaceX
Success

Success
81
17 February 2020
15:05
F9 B5
CCAFS
Starlink
Starlink
LEO
SpaceX
Success

Failure
82
7 March 2020
04:50
F9 B5
CCAFS

SpaceX CRS-20
SpaceX CRS-20
LEO
NASA (CRS)
Success

Success
83
18 March 2020
12:16
F9 B5
KSC
Starlink
Starlink
LEO
SpaceX
Success

Failure
84
22 April 2020
19:30
F9 B5
KSC
Starlink
Starlink
LEO
SpaceX
Success

Success
85
30 May 2020
19:22
F9 B5
KSC
Crew Dragon Demo-2
Crew Dragon Demo-2
LEO
NASA (CCDev)
Success

Success
86
4 June 2020
01:25
F9 B5
CCAFS
Starlink
Starlink
LEO
SpaceX
Success

Success
87
13 June 2020
09:21
F9 B5
CCAFS

Starlink
Starlink
LEO
SpaceXPlanet Labs
Success

Success
88
30 June 2020
20:10:46
F9 B5
CCAFS
GPS III
GPS III
MEO
U.S. Space Force[530]
Success

Success
89
20 July 2020
21:30
F9 B5B1058.2
CCAFS
ANASIS-II
ANASIS-II
GTO
Republic of Korea Army
Success

Success
90
7 August 2020
05:12
F9 B5
KSC
Starlink
Starlink
LEO
SpaceXSpaceflight Industries (BlackSky)
Success

Success
91
18 August 2020
14:31
F9 B5B1049.6
CCAFS
Starlink
Starlink
LEO
SpaceXPlanet Labs
Success

Success
92
30 August 2020
23:18
F9 B5
CCAFS

SAOCOM 1B
SAOCOM 1B
SSO
CONAEPlanetIQTyvak
Success

Success
93
3 September 2020
12:46:14
F9 B5B1060.2
KSC
Starlink
Starlink
LEO
SpaceX
Success

Success
94
6 October 2020
11:29:34
F9 B5B1058.3
KSC
Starlink
Starlink
LEO
SpaceX
Success

Success
95
18 October 2020
12:25:57
F9 B5B1051.6
KSC
Starlink
Starlink
LEO
SpaceX
Success

Success
96
24 October 2020
15:31:34
F9 B5
CCAFS
Starlink
Starlink
LEO
SpaceX
Success

Success
97
5 November 2020
23:24:23
F9 B5
CCAFS

GPS III
GPS III
MEO
USSF
Success

Success
98
16 November 2020
00:27
F9 B5
KSC
Crew-1
Crew-1
LEO
NASA (CCP)[497]
Success

Success
99
21 November 2020
17:17:08
F9 B5
VAFB
Sentinel-6 Michael Freilich (Jason-CS A)
Sentinel-6 Michael Freilich (Jason-CS A)
LEO
NASA / NOAA / ESA / EUMETSAT
Success

Success
100
25 November 2020
02:13
F9 B5 ♺
CCAFS
Starlink
Starlink
LEO
SpaceX
Success

Success
101
6 December 2020
16:17:08
F9 B5 ♺
KSC
SpaceX CRS-21
SpaceX CRS-21
LEO
NASA (CRS)
Success

Success
102
13 December 2020
17:30:00
F9 B5 ♺
CCSFS

SXM-7
SXM-7
GTO
Sirius XM
Success

Success
103
19 December 2020
14:00:00
F9 B5 ♺
KSC
NROL-108
NROL-108
LEO
NRO
Success

Success
104
8 January 2021
02:15
F9 B5
CCSFS
Türksat 5A
Türksat 5A
GTO
Türksat
Success

Success
105
20 January 2021
13:02
F9 B5B1051.8
KSC
Starlink
Starlink
LEO
SpaceX
Success

Success
106
24 January 2021
15:00
F9 B5B1058.5
CCSFS
Transporter-1
Transporter-1
SSO
Various
Success

Success
107
4 February 2021
06:19
F9 B5 ♺
CCSFS

Starlink
Starlink
LEO
SpaceX
Success

Success
108
16 February 2021
03:59:37
F9 B5 ♺
CCSFS
Starlink
Starlink
LEO
SpaceX
Success

Failure
109
4 March 2021
08:24
F9 B5 ♺
KSC
Starlink
Starlink
LEO
SpaceX
Success

Success
110
11 March 2021
08:13:29
F9 B5 ♺
CCSFS
Starlink
Starlink
LEO
SpaceX
Success

Success
111
14 March 2021
10:01
F9 B5 ♺
KSC
Starlink
Starlink
LEO
SpaceX
Success

Success
112
24 March 2021
08:28
F9 B5B1060.6
CCSFS

Starlink
Starlink
LEO
SpaceX
Success

Success
113
7 April 2021
16:34
F9 B5 ♺
CCSFS
Starlink
Starlink
LEO
SpaceX
Success

Success
114
23 April 2021
9:49
F9 B5B1061.2
KSC
Crew-2
Crew-2
LEO
NASA (CTS)[497]
Success

Success
115
29 April 2021
03:44
F9 B5B1060.7
CCSFS
Starlink
Starlink
LEO
SpaceX
Success

Success
116
4 May 2021
19:01
F9 B5B1049.9
KSC
Starlink
Starlink
LEO
SpaceX
Success

Success
117
9 May 2021
06:42
F9 B5B1051.10
CCSFS

```
Starlink
Starlink
LEO
SpaceX
Success

Success
118
15 May 2021
22:56
F9 B5B1058.8
KSC
Starlink
Starlink
LEO
SpaceX Capella Space and Tyvak
Success

Success
119
26 May 2021
18:59
F9 B5B1063.2
CCSFS
Starlink
Starlink
LEO
SpaceX
Success

Success
120
3 June 2021
17:29
F9 B5B1067.1
KSC
SpaceX CRS-22
SpaceX CRS-22
LEO
NASA (CRS)
Success

Success
121
6 June 2021
04:26
F9 B5
CCSFS
SXM-8
SXM-8
GTO
Sirius XM
Success

Success
```

After you have fill in the parsed launch record values into `launch_dict` , you can create a dataframe from it.

In [40]:
```
df=pd.DataFrame(launch_dict)
```

In [41]:
```
df
```

Out[41]:

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	
1	2	CCAFS	Dragon	0	LEO	NASA (COTS)\nNRO	Success	F9 v1.0B0004.1	Failure	[
2	3	CCAFS	Dragon	525 kg	LEO	NASA (COTS)	Success	F9 v1.0B0005.1	No attempt\n	
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA (CRS)	Success\n	F9 v1.0B0006.1	No attempt	8
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA (CRS)	Success\n	F9 v1.0B0007.1	No attempt\n	
...	...	...	...	...	...	...	...	...	...	
116	117	CCSFS	Starlink	15,600 kg	LEO	SpaceX	Success\n	F9 B5B1051.10	Success	
117	118	KSC	Starlink	~14,000 kg	LEO	SpaceX Capella Space and Tyvak	Success\n	F9 B5B1058.8	Success	
118	119	CCSFS	Starlink	15,600 kg	LEO	SpaceX	Success\n	F9 B5B1063.2	Success	
119	120	KSC	SpaceX CRS-22	3,328 kg	LEO	NASA (CRS)	Success\n	F9 B5B1067.1	Success	
120	121	CCSFS	SXM-8	7,000 kg	GTO	Sirius XM	Success\n	F9 B5	Success	

121 rows × 11 columns

We can now export it to a **CSV** for the next section, but to make the answers consistent and in case you have difficulties finishing this lab.

Following labs will be using a provided dataset to make each lab independent.

```
df.to_csv('spacex_web_scraped.csv', index=False)
```

Authors

Yan Luo

Nayef Abou Tayoun

Change Log

Date (YYYY-MM-DD)	Version	Changed By	Change Description
2021-06-09	1.0	Yan Luo	Tasks updates
2020-11-10	1.0	Nayef	Created the initial version