Phase	Activity	Code and Output
| Extract | Loading the dataset | ```python
# Loading the dataset
import pandas as pd


df = pd.read_csv('RT_IOT2022.csv')
``` |
| Extract | Checking Column Names | ```python
df.columns
```
```
Index(['no', 'id.orig_p', 'id.resp_p', 'proto', 'service', 'flow_duration',
       'fwd_pkts_tot', 'bwd_pkts_tot', 'fwd_data_pkts_tot',
       'bwd_data_pkts_tot', 'fwd_pkts_per_sec', 'bwd_pkts_per_sec',
       'flow_pkts_per_sec', 'down_up_ratio', 'fwd_header_size_tot',
       'fwd_header_size_min', 'fwd_header_size_max', 'bwd_header_size_tot',
       'bwd_header_size_min', 'bwd_header_size_max', 'flow_FIN_flag_count',
       'flow_SYN_flag_count', 'flow_RST_flag_count', 'fwd_PSH_flag_count',
       'bwd_PSH_flag_count', 'flow_ACK_flag_count', 'fwd_URG_flag_count',
       'bwd_URG_flag_count', 'flow_CWR_flag_count', 'flow_ECE_flag_count',
       'fwd_pkts_payload.min', 'fwd_pkts_payload.max', 'fwd_pkts_payload.tot',
       'fwd_pkts_payload.avg', 'fwd_pkts_payload.std', 'bwd_pkts_payload.min',
       'bwd_pkts_payload.max', 'bwd_pkts_payload.tot', 'bwd_pkts_payload.avg',
       'bwd_pkts_payload.std', 'flow_pkts_payload.min',
       'flow_pkts_payload.max', 'flow_pkts_payload.tot',
       'flow_pkts_payload.avg', 'flow_pkts_payload.std', 'fwd_iat.min',
       'fwd_iat.max', 'fwd_iat.tot', 'fwd_iat.avg', 'fwd_iat.std',
       'bwd_iat.min', 'bwd_iat.max', 'bwd_iat.tot', 'bwd_iat.avg',
       'bwd_iat.std', 'flow_iat.min', 'flow_iat.max', 'flow_iat.tot',
       'flow_iat.avg', 'flow_iat.std', 'payload_bytes_per_second',
       'fwd_subflow_pkts', 'bwd_subflow_pkts', 'fwd_subflow_bytes',
       'bwd_subflow_bytes', 'fwd_bulk_bytes', 'bwd_bulk_bytes',
       'fwd_bulk_packets', 'bwd_bulk_packets', 'fwd_bulk_rate',
       'bwd_bulk_rate', 'active.min', 'active.max', 'active.tot', 'active.avg',
       'active.std', 'idle.min', 'idle.max', 'idle.tot', 'idle.avg',
       'idle.std', 'fwd_init_window_size', 'bwd_init_window_size',
       'fwd_last_window_size', 'Attack_type'],
      dtype='object')
``` |
| Extract | Checking for unique Values | ```python
df['proto'].unique()
```
```
array(['tcp', 'udp', 'icmp'], dtype=object)
``` |

		```df['service'].unique()```
		
		```array(['mqtt', '-', 'http', 'dns', 'ntp', 'ssl', 'dhcp', 'irc', 'ssh', 'radius'], dtype=object)```
		
		```df['Attack_type'].unique()```
		
		```array(['MQTT_Publish', 'Thing_Speak', 'Wipro_bulb', 'ARP_poisioning', 'DDOS_Slowloris', 'DOS_SYN_Hping', 'Metasploit_Brute_Force_SSH', 'NMAP_FIN_SCAN', 'NMAP_OS_DETECTION', 'NMAP_TCP_scan', 'NMAP_UDP_SCAN', 'NMAP_XMAS_TREE_SCAN'], dtype=object)```

Extract	Checking for null entries	```[4]: df.isnull().sum()```
		
		```[4]: no 0```
		``` id.orig_p 0```
		``` id.resp_p 0```
		``` proto 0```
		``` service 0```
		``` ..```
		``` idle.std 0```
		``` fwd_init_window_size 0```
		``` bwd_init_window_size 0```
		``` fwd_last_window_size 0```
		``` Attack_type 0```
		``` Length: 85, dtype: int64```

Extract	Invalid Entries	```[1]: (df['service'] == '-').count()```
		
		```[1]: np.int64(123117)```

Extract	Check DDOS attacks	```dos = df[df['Attack_type'] == 'DDOS_Slowloris']``` <table><tr><th>uration</th><th>fwd_pkts_tot</th><th>bwd_pkts_tot</th><th>fwd_data_pkts_tot</th><th>bwd_data_pkts_tot</th><th>...</th><th>active.std</th><th>idle.min</th><th>idle.max</th><th>idle.tot</th><th>idle.avg</th><th>idle.std</th><th>fwd_init_window_size</th><th>bwd_init_window_size</th><th>fwd_last_window_size</th><th>Attack_type</th></tr><tr><td>.000181</td><td>8</td><td>0</td><td>2</td><td>0</td><td>...</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>65476</td><td>0</td><td>0</td><td>DDOS_Slowloris</td></tr><tr><td>.000536</td><td>8</td><td>0</td><td>2</td><td>0</td><td>...</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>65476</td><td>0</td><td>0</td><td>DDOS_Slowloris</td></tr><tr><td>.000551</td><td>8</td><td>0</td><td>2</td><td>0</td><td>...</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>65476</td><td>0</td><td>0</td><td>DDOS_Slowloris</td></tr><tr><td>.000546</td><td>8</td><td>0</td><td>2</td><td>0</td><td>...</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>65476</td><td>0</td><td>0</td><td>DDOS_Slowloris</td></tr><tr><td>.000539</td><td>8</td><td>0</td><td>2</td><td>0</td><td>...</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>65476</td><td>0</td><td>0</td><td>DDOS_Slowloris</td></tr><tr><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td></tr><tr><td>.003119</td><td>4</td><td>3</td><td>2</td><td>0</td><td>...</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>64240</td><td>65160</td><td>502</td><td>DDOS_Slowloris</td></tr><tr><td>.003120</td><td>4</td><td>3</td><td>2</td><td>0</td><td>...</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>64240</td><td>65160</td><td>502</td><td>DDOS_Slowloris</td></tr><tr><td>.004047</td><td>4</td><td>3</td><td>2</td><td>0</td><td>...</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>64240</td><td>65160</td><td>502</td><td>DDOS_Slowloris</td></tr><tr><td>.003871</td><td>4</td><td>3</td><td>2</td><td>0</td><td>...</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>64240</td><td>65160</td><td>502</td><td>DDOS_Slowloris</td></tr><tr><td>.000000</td><td>1</td><td>0</td><td>1</td><td>0</td><td>...</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0</td><td>0</td><td>0</td><td>DDOS_Slowloris</td></tr></table>

Extract		
Transform	Rounding off values	
Load		

For the Extract cell:

```
df[df['fwd_bulk_packets'] > 0]
```

	no	id.orig_p	id.resp_p	proto	service	flow_duration	fwd_pkts_tot	bwd_pkts_tot	fwd_data_pkts_tot	bwd_data_pkts_tot	...	active.std	idle.min	idle.max	idle.tot	idle.avg	
4520	374	52375	443	tcp	ssl	21.379392	92	112	11	99	...	3.465367e+06	1.942203e+07	1.942203e+07	1.942203e+07	1.942203e+07	0.000
4535	389	52372	443	tcp	ssl	11.049312	40	39	20	19	...	8.635407e+05	5.658469e+06	5.658469e+06	5.658469e+06	5.658469e+06	0.000
4536	390	52374	443	tcp	ssl	1.106636	19	17	8	9	...	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000
4540	394	52370	443	tcp	ssl	65.690455	18	21	8	12	...	1.016011e+06	6.425345e+07	6.425345e+07	6.425345e+07	6.425345e+07	0.000
4602	456	58116	443	tcp	ssl	1.100566	32	34	9	21	...	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
20224	7717	48848	443	tcp	ssl	184.457190	56	42	43	9	...	1.330593e+05	1.364865e+07	5.891797e+07	1.836208e+08	4.590520e+07	2.170
20227	7720	57772	443	tcp	ssl	171.688333	31	26	17	12	...	4.422009e+05	5.349604e+07	5.893790e+07	1.706322e+08	5.687739e+07	2.951
20639	382	49634	80	tcp	http	26.774708	11	1	9	0	...	0.000000e+00	6.719394e+06	1.343429e+07	2.015369e+07	1.007684e+07	4.748
121105	2588	3	3	icmp	-	883.106953	882	0	882	0	...	1.767556e+08	8.120829e+06	8.120829e+06	8.120829e+06	8.120829e+06	1.837
121106	2589	3	3	icmp	-	905.964201	903	0	903	0	...	4.774862e+08	6.412426e+06	9.010986e+06	1.542341e+07	7.711706e+06	1.837

246 rows × 85 columns

For the Transform cell:

```
df = round(df['flow_duration'], 6)
df.head()

0    32.011598
1    31.883584
2    32.124053
3    31.961063
4    31.902362
Name: flow_duration, dtype: float64
```