In [3]:
```
import pandas as pd
```

In [5]:
```
bombs=pd.read_excel(r'C:\Venkat\Python\Practice_Material\PandasPractice.xlsx')
```

In [11]:
```
bombs.columns
```

Out[11]:
```
Index(['S.No', 'NAME OF THE PRODUCTS', 'Unnamed: 2', 'MARKET RATE',
       'SALE RATE', 'PER', 'QTY', 'AMOUNT'],
      dtype='object')
```

In [23]:
```
bombs[[bombs['S.No']]]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[23], line 1
----> 1 bombs[[bombs['S.No']]]

File ~\anaconda3\Lib\site-packages\pandas\core\frame.py:4108, in DataFrame.__getitem__(self, key)
   4106     if is_iterator(key):
   4107         key = list(key)
-> 4108     indexer = self.columns._get_indexer_strict(key, "columns")[1]
   4110 # take() does not accept boolean indexers
   4111 if getattr(indexer, "dtype", None) == bool:

File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:6200, in Index._get_indexer_strict(self, key, axis_name)
   6197 else:
   6198     keyarr, indexer, new_indexer = self._reindex_non_unique(keyarr)
-> 6200 self._raise_if_missing(keyarr, indexer, axis_name)
   6202 keyarr = self.take(indexer)
   6203 if isinstance(key, Index):
   6204     # GH 42790 - Preserve name from an Index

File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:6249, in Index._raise_if_missing(self, key, indexer, axis_name)
   6247 if nmissing:
   6248     if nmissing == len(indexer):
-> 6249         raise KeyError(f"None of [{key}] are in the [{axis_name}]")
   6251     not_found = list(ensure_index(key)[missing_mask.nonzero()[0]].unique())
   6252     raise KeyError(f"{not_found} not in index")

KeyError: "None of [Index([(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72)], dtype='object')] are in the [columns]"
```

In [9]:
```
bombs.isnull
```

Out[9]:
```
<bound method DataFrame.isnull of       S.No    NAME OF THE PRODUCTS Unnamed: 2 MARKET RATE  Unnamed: 4  \
0      1.0              4" Lakshmi        NaN         100         NaN
1      2.0         4" Gold Lakshmi        NaN         150         NaN
2      3.0           Little Singam        NaN         150         NaN
3      4.0                   Kumki        NaN         200         NaN
4      5.0               Bhabhubali        NaN         260         NaN
..     ...                      ...        ...         ...         ...
133  126.0     30 Cm Green Sparkle        NaN         NaN       205.0
134  127.0       30 Cm Red Sparkle        NaN         NaN       210.0
135  128.0  50 Cm Electric Sparkle        NaN         NaN       760.0
136  129.0    50 Cm Colour Sparkle        NaN         NaN       935.0
137  130.0        Rotating Sparkle        NaN         NaN      1230.0

     SALE RATE    PER  QTY  AMOUNT
0           20  1 PKT  NaN       0
1           30  1 PKT  NaN       0
2           30  1 PKT  NaN       0
3           40  1 PKT  NaN       0
4           52  1 PKT  NaN       0
..         ...    ...  ...     ...
133         41  1 BOX  NaN       0
134         42  1 BOX  NaN       0
135        152  1 BOX  NaN       0
136        187  1 BOX  4.0     748
137        246  1 BOX  4.0     984

[138 rows x 9 columns]>
```

In [5]:
```
bombs_1=pd.read_excel(r'C:\Venkat\Python\Practice_Material\PandasPracticeNotNull.xlsx')
```

In [7]:
```
bombs_1
```

Out[7]:

	S.No	NAME OF THE PRODUCTS	MARKET RATE	SALE RATE	PER	QTY	AMOUNT
0	1	4" Lakshmi	100	20	1 PKT	0	0
1	2	4" Gold Lakshmi	150	30	1 PKT	0	0
2	3	Little Singam	150	30	1 PKT	0	0
3	4	Kumki	200	40	1 PKT	0	0
4	5	Bhabhubali	260	52	1 PKT	0	0
...	...	...	...	...	...	...	...
67	68	Siren	1000	200	1 BOX	0	0
68	69	Smoke Stick	100	20	1 BOX	0	0
69	70	Colour Smoke	900	180	1 BOX	0	0
70	71	Sound Marriage	800	160	1 BOX	0	0
71	72	Lolly Pop	1500	300	1 BOX	0	0

72 rows × 7 columns

In [27]:
```
bombs_1.columns
```

Out[27]:
```
Index(['S.No', 'NAME OF THE PRODUCTS', 'MARKET RATE', 'SALE RATE', 'PER',
       'QTY', 'AMOUNT'],
      dtype='object')
```

In [31]:
```
bombs_1.values
```

Out[31]:
```
array([[1, '4" Lakshmi', 100, 20, '1 PKT', 0, 0],
       [2, '4" Gold Lakshmi', 150, 30, '1 PKT', 0, 0],
       [3, 'Little Singam', 150, 30, '1 PKT', 0, 0],
       [4, 'Kumki', 200, 40, '1 PKT', 0, 0],
       [5, 'Bhabhubali', 260, 52, '1 PKT', 0, 0],
       [6, 'Jallikattu', 300, 60, '1 PKT', 0, 0],
       [7, '2 3/4" Kuruvi', 50, 10, '1 PKT', 0, 0],
       [8, '2 Sound', 200, 40, '1 PKT', 0, 0],
       [9, 'Flower Pots Small', 320, 64, '1 BOX', 0, 0],
       [10, 'Flower Pots Big', 400, 80, '1 BOX', 0, 0],
       [11, 'Flower Pots Special', 450, 90, '1 BOX', 0, 0],
       [12, 'Flower Pots Asoka', 620, 124, '1 BOX', 0, 0],
       [13, 'Colour Kotti', 1160, 232, '1 BOX', 2, 464],
       [14, 'Colour Kotti DLX', 1600, 320, '1 BOX', 2, 640],
       [15, 'Chakka r Big (10 Pcs)', 200, 40, '1 BOX', 0, 0],
       [16, 'Chakka r Big  (25 Pcs)', 500, 100, '1 BOX', 0, 100],
       [17, 'Chakka r Asoka', 440, 88, '1 BOX', 0, 0],
       [18, 'Chakka r Speci al', 480, 96, '1 BOX', 0, 0],
       [19, 'Spinner Speci al', 680, 136, '1 BOX', 0, 0],
       [20, 'Chakka r Dlx', 680, 160, '1 BOX', 2, 320],
       [21, 'Lotus', 1460, 292, '1 BOX', 2, 584],
       [22, "1¼' Twinkling Star Small", 120, 24, '1 BOX', 0, 0],
       [23, "4' Twinkling Star Dlx", 350, 70, '1 BOX', 0, 0],
       [24, 'Red Bijili', 200, 40, '1 PKT', 0, 0],
       [25, 'Striped  Bijili', 250, 50, '1 PKT', 0, 0],
       [26, 'Super DLX', 400, 80, '1 BOX', 0, 0],
       [27, 'Flash', 300, 60, '1 BOX', 0, 0],
       [28, 'Pock Man', 800, 160, '1 BOX', 0, 0],
       [29, 'Mega DLX 5 in 1', 800, 160, '1 BOX', 0, 0],
       [30, 'Mega DLX 10 in 1', 1250, 250, '1 BOX', 0, 0],
       [31, 'Auto  Bomb', 250, 50, '1 BOX', 0, 0],
       [32, 'Hydro  Bomb', 350, 70, '1 BOX', 0, 0],
       [33, 'King of King Small', 420, 84, '1 BOX', 0, 0],
       [34, 'Classic Bomb', 520, 104, '1 BOX', 0, 0],
       [35, 'Digital Bomb', 950, 190, '1 BOX', 0, 0],
       [36, 'Hollywood Bomb', 1100, 220, '1 BOX', 0, 0],
       [37, 'Water Fall - 5 Pcs', 1100, 220, '1 PKT', 0, 0],
       [38, 'Nayagara Falls - 3 Pcs', 600, 120, '1 PKT', 0, 0],
       [39, 'Candy Crash', 550, 110, '1 PKT', 0, 0],
       [40, '28 Chorsa', 90, 18, '1 PKT', 0, 0],
       [41, '28 Giant', '140. 00', 28, '1 PKT', 0, 0],
       [42, '56 Giant', '280. 00', 56, '1 PKT', 0, 0],
       [43, '24 Dlx', '250. 00', 50, '1 PKT', 0, 0],
       [44, '50 Dlx', '550. 00', 110, '1 PKT', 0, 0],
       [45, '100 Dlx', '1100. 00', '220. 00', '1 PKT', 0, 0],
       [46, '100 Wala', 170, 34, '1 BOX', 0, 0],
       [47, '200 Wala', 340, 68, '1 BOX', 0, 0],
       [48, '1000 Wala', 1000, 200, '1 BOX', 0, 0],
       [49, '2000 Wala', 2000, 400, '1 BOX', 0, 0],
       [50, '5000 Wala', 5000, 1000, '1 BOX', 0, 0],
       [51, '10000 Wala', 10000, 2000, '1 BOX', 0, 0],
       [52, 'Adiyal Paper Bomb 1/4 Kg', 300, 60, '1 BOX', 0, 0],
       [53, 'Adiyal Paper Bomb 1/2 Kg', 600, 120, '1 BOX', 0, 0],
       [54, 'Adiyal Paper Bomb 1 Kg', 1200, 240, '1 BOX', 0, 480],
       [55, 'Rocket Bombs', 350, 70, '1 BOX', 0, 0],
       [56, 'Lunic Rocket', 700, 140, '1 BOX', 0, 0],
       [57, 'Two Sound Rocket', 760, 152, '1 BOX', 0, 304],
       [58, 'Whistling Rocket', 1000, 200, '1 BOX', 2, 400],
       [59, 'Butterfly', 600, 120, '1 BOX', 2, 240],
       [60, 'Helicopter', 500, 100, '1 BOX', 2, 200],
       [61, 'Drone', 900, 180, '1 BOX', 0, 0],
       [62, 'Tom & Jerry', 300, 60, '1 BOX', 0, 0],
       [63, 'Roll Cap', 400, 80, '1 BOX', 0, 0],
       [64, 'Top Gun', 1200, 240, '1 BOX', 0, 0],
       [65, 'Snake Tablet', 100, 20, '1 BOX', 0, 0],
       [66, 'Pop & Pop', 50, 10, '1 BOX', 0, 0],
       [67, 'Money Bank', 1000, 200, '1 BOX', 0, 0],
       [68, 'Siren', 1000, 200, '1 BOX', 0, 0],
       [69, 'Smoke Stick', 100, 20, '1 BOX', 0, 0],
       [70, 'Colour Smoke', 900, 180, '1 BOX', 0, 0],
       [71, 'Sound Marriage', 800, 160, '1 BOX', 0, 0],
       [72, 'Lolly Pop', 1500, 300, '1 BOX', 0, 0]], dtype=object)
```

In [39]:
```
bombs_1.count()
```

Out[39]:
```
S.No                    72
NAME OF THE PRODUCTS    72
MARKET RATE             72
SALE RATE               72
PER                     72
QTY                     72
AMOUNT                  72
dtype: int64
```

In [47]:
```
bombs_1.shape
```

Out[47]:
```
(72, 7)
```

In [57]:
```
bombs_1[:,1:]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3805, in Index.get_loc(self, key)
   3804 try:
-> 3805     return self._engine.get_loc(casted_key)
   3806 except KeyError as err:

File index.pyx:167, in pandas._libs.index.IndexEngine.get_loc()

File index.pyx:196, in pandas._libs.index.IndexEngine.get_loc()

File pandas\\_libs\\hashtable_class_helper.pxi:7081, in pandas._libs.hashtable.PyObjectHashTable.get_item()

File pandas\\_libs\\hashtable_class_helper.pxi:7089, in pandas._libs.hashtable.PyObjectHashTable.get_item()

KeyError: (slice(None, None, None), slice(1, None, None))

During handling of the above exception, another exception occurred:

InvalidIndexError                         Traceback (most recent call last)
Cell In[57], line 1
----> 1 bombs_1[:,1:]

File ~\anaconda3\Lib\site-packages\pandas\core\frame.py:4102, in DataFrame.__getitem__(self, key)
   4100 if self.columns.nlevels > 1:
   4101     return self._getitem_multilevel(key)
-> 4102 indexer = self.columns.get_loc(key)
   4103 if is_integer(indexer):
   4104     indexer = [indexer]

File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3811, in Index.get_loc(self, key)
   3806 except KeyError as err:
   3807     if isinstance(casted_key, slice) or (
   3808         isinstance(casted_key, abc.Iterable)
   3809         and any(isinstance(x, slice) for x in casted_key)
   3810     ):
-> 3811         raise InvalidIndexError(key)
   3812     raise KeyError(key) from err
   3813 except TypeError:
   3814     # If we have a listlike key, _check_indexing_error will raise
   3815     #  InvalidIndexError. Otherwise we fall through and re-raise
   3816     #  the TypeError.

InvalidIndexError: (slice(None, None, None), slice(1, None, None))
```

In [59]:
```
bombs_1.ndim
```

Out[59]: 2

In [71]:
```

```

Out[71]:

	S.No	NAME OF THE PRODUCTS	MARKET RATE	SALE RATE	PER	QTY	AMOUNT
1	2	4" Gold Lakshmi	150	30	1 PKT	0	0

In [65]:
```
bombs_1[:0,:0]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3805, in Index.get_loc(self, key)
   3804 try:
-> 3805     return self._engine.get_loc(casted_key)
   3806 except KeyError as err:

File index.pyx:167, in pandas._libs.index.IndexEngine.get_loc()

File index.pyx:196, in pandas._libs.index.IndexEngine.get_loc()

File pandas\\_libs\\hashtable_class_helper.pxi:7081, in pandas._libs.hashtable.PyObjectHashTable.get_item()

File pandas\\_libs\\hashtable_class_helper.pxi:7089, in pandas._libs.hashtable.PyObjectHashTable.get_item()

KeyError: (slice(None, 0, None), slice(None, 0, None))

During handling of the above exception, another exception occurred:

InvalidIndexError                         Traceback (most recent call last)
Cell In[65], line 1
----> 1 bombs_1[:0,:0]

File ~\anaconda3\Lib\site-packages\pandas\core\frame.py:4102, in DataFrame.__getitem__(self, key)
   4100 if self.columns.nlevels > 1:
   4101     return self._getitem_multilevel(key)
-> 4102 indexer = self.columns.get_loc(key)
   4103 if is_integer(indexer):
   4104     indexer = [indexer]

File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3811, in Index.get_loc(self, key)
   3806 except KeyError as err:
   3807     if isinstance(casted_key, slice) or (
   3808         isinstance(casted_key, abc.Iterable)
   3809         and any(isinstance(x, slice) for x in casted_key)
   3810     ):
-> 3811         raise InvalidIndexError(key)
   3812     raise KeyError(key) from err
   3813 except TypeError:
   3814     # If we have a listlike key, _check_indexing_error will raise
   3815     #  InvalidIndexError. Otherwise we fall through and re-raise
   3816     #  the TypeError.

InvalidIndexError: (slice(None, 0, None), slice(None, 0, None))
```

In [73]:
```
bombs_1[0:1]
```

Out[73]:

	S.No	NAME OF THE PRODUCTS	MARKET RATE	SALE RATE	PER	QTY	AMOUNT
0	1	4" Lakshmi	100	20	1 PKT	0	0

In [75]:
```
bombs_1[0]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3805, in Index.get_loc(self, key)
   3804 try:
-> 3805     return self._engine.get_loc(casted_key)
   3806 except KeyError as err:

File index.pyx:167, in pandas._libs.index.IndexEngine.get_loc()

File index.pyx:196, in pandas._libs.index.IndexEngine.get_loc()

File pandas\\_libs\\hashtable_class_helper.pxi:7081, in pandas._libs.hashtable.PyObjectHashTable.get_item()

File pandas\\_libs\\hashtable_class_helper.pxi:7089, in pandas._libs.hashtable.PyObjectHashTable.get_item()

KeyError: 0

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
Cell In[75], line 1
----> 1 bombs_1[0]

File ~\anaconda3\Lib\site-packages\pandas\core\frame.py:4102, in DataFrame.__getitem__(self, key)
   4100 if self.columns.nlevels > 1:
   4101     return self._getitem_multilevel(key)
-> 4102 indexer = self.columns.get_loc(key)
   4103 if is_integer(indexer):
   4104     indexer = [indexer]

File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3812, in Index.get_loc(self, key)
   3807     if isinstance(casted_key, slice) or (
   3808         isinstance(casted_key, abc.Iterable)
   3809         and any(isinstance(x, slice) for x in casted_key)
   3810     ):
   3811         raise InvalidIndexError(key)
-> 3812     raise KeyError(key) from err
   3813 except TypeError:
   3814     # If we have a listlike key, _check_indexing_error will raise
   3815     #  InvalidIndexError. Otherwise we fall through and re-raise
   3816     #  the TypeError.
   3817     self._check_indexing_error(key)

KeyError: 0
```

In [77]:
```
bombs_1[1]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3805, in Index.get_loc(self, key)
   3804 try:
-> 3805     return self._engine.get_loc(casted_key)
   3806 except KeyError as err:

File index.pyx:167, in pandas._libs.index.IndexEngine.get_loc()

File index.pyx:196, in pandas._libs.index.IndexEngine.get_loc()

File pandas\\_libs\\hashtable_class_helper.pxi:7081, in pandas._libs.hashtable.PyObjectHashTable.get_item()

File pandas\\_libs\\hashtable_class_helper.pxi:7089, in pandas._libs.hashtable.PyObjectHashTable.get_item()

KeyError: 1

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
Cell In[77], line 1
----> 1 bombs_1[1]

File ~\anaconda3\Lib\site-packages\pandas\core\frame.py:4102, in DataFrame.__getitem__(self, key)
   4100 if self.columns.nlevels > 1:
   4101     return self._getitem_multilevel(key)
-> 4102 indexer = self.columns.get_loc(key)
   4103 if is_integer(indexer):
   4104     indexer = [indexer]

File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3812, in Index.get_loc(self, key)
   3807     if isinstance(casted_key, slice) or (
   3808         isinstance(casted_key, abc.Iterable)
   3809         and any(isinstance(x, slice) for x in casted_key)
   3810     ):
   3811         raise InvalidIndexError(key)
-> 3812     raise KeyError(key) from err
   3813 except TypeError:
   3814     # If we have a listlike key, _check_indexing_error will raise
   3815     #  InvalidIndexError. Otherwise we fall through and re-raise
   3816     #  the TypeError.
   3817     self._check_indexing_error(key)

KeyError: 1
```

In [89]:
```
bombs_1['S.No']
```

Out[89]:
```
0      1
1      2
2      3
3      4
4      5
      ..
67    68
68    69
69    70
70    71
71    72
Name: S.No, Length: 72, dtype: int64
```

In [91]:
```
bombs_1['S.No'].shape
```

Out[91]:
```
(72,)
```

In [95]:
```
bombs_1.columns
```

Out[95]:
```
Index(['S.No', 'NAME OF THE PRODUCTS', 'MARKET RATE', 'SALE RATE', 'PER',
       'QTY', 'AMOUNT'],
      dtype='object')
```

In [107…
```
bombs_1[:1]
```

Out[107…

	S.No	NAME OF THE PRODUCTS	MARKET RATE	SALE RATE	PER	QTY	AMOUNT
0	1	4" Lakshmi	100	20	1 PKT	0	0

In [9]:
```
bombs_1.shape
```

Out[9]: (72, 7)

In [15]:
```
#bombs_2=pd.read_excel(open(r'C:\Venkat\Python\Practice_Material\PandasPracticeNotNull.xlsx'),sheet_name='Sheet2')
```

In [19]:
```
bombs_1[['S.No']]
```

Out[19]:

	S.No
0	1
1	2
2	3
3	4
4	5
...	...
67	68
68	69
69	70
70	71
71	72

72 rows × 1 columns

In [29]:
```
bombs_1[bombs_1['NAME OF THE PRODUCTS']=='Siren']
```

Out[29]:

	S.No	NAME OF THE PRODUCTS	MARKET RATE	SALE RATE	PER	QTY	AMOUNT
67	68	Siren	1000	200	1 BOX	0	0

In [25]:
```
bombs['S.No']
```

Out[25]:
```
0      1
1      2
2      3
3      4
4      5
      ..
67    68
68    69
69    70
70    71
71    72
Name: S.No, Length: 72, dtype: int64
```

In [ ]: