



# Knowledge Check

## Knowledge Check

1

How can you access the column *Age* in a Pandas DataFrame named *df*?

- A. `df('Age')`
- B. `df[0]`
- C. `df['Age']`
- D. `df.Age()`



## Knowledge Check

1

How can you access the column 'Age' in a Pandas DataFrame named 'df'?

- A. `df('Age')`
- B. `df[0]`
- C. `df['Age']`
- D. `df.Age()`

---

The correct answer is **C**

---

In Pandas, DataFrame columns can be accessed using the square bracket notation with the column name as a string.



**Knowledge  
Check**

**2**

**Which Pandas function is used to obtain a summary of descriptive statistics for a DataFrame named *df*?**

- A. `df.describe()`
- B. `df.statistics()`
- C. `df.summary()`
- D. `df.info()`



## Knowledge Check

2

Which Pandas function is used to obtain a summary of descriptive statistics for a DataFrame named *df*?

- A. `df.describe()`
- B. `df.statistics()`
- C. `df.summary()`
- D. `df.info()`

---

The correct answer is **A**

---

The `describe()` function in Pandas is utilized for providing a summary of descriptive statistics, including measures such as mean, median, and standard deviation for numeric columns.



**Knowledge  
Check**  
**3**

**How is the year extracted from a Pandas Series *date\_series* containing datetime objects?**

- A. `date_series.year()`
- B. `date_series.get('year')`
- C. `date_series.dt.year`
- D. `year(date_series)`



Knowledge  
Check  
3

How is the year extracted from a Pandas Series *date\_series* containing datetime objects?

- A. `date_series.year()`
- B. `date_series.get('year')`
- C. `date_series.dt.year`
- D. `year(date_series)`

---

The correct answer is **C**

---

Pandas uses the `dt` accessor to access the datetime properties of a Series, with `.year` specifically extracting the year component from each datetime object in the series.





**Thank You**