UnCodemy

Pandas Interview Questions PDF

Pandas interviews test creating and inspecting DataFrames, selecting with loc and iloc, filtering, groupby and aggregation, merging, handling missing values, working with dates and strings, pivoting, and basic performance. This page gives short model answers with code you can run on a small made-up table.

Key takeaways

- Know the DataFrame and Series, and the difference between loc (labels) and iloc (positions).
- groupby with agg is the core summarising tool; merge is the core joining tool.
- Handle missing data deliberately with isna, fillna and dropna.
- Use vectorised operations instead of loops for speed.
- Check shapes and row counts after every merge or filter.

What is pandas interview questions?

Pandas is a Python library for data analysis that provides the DataFrame, a table of rows and labelled columns, along with tools to clean, reshape, combine, group and summarise data.

What data do the examples use?

The examples use a small made-up table of orders. Run this first to follow along.

```
import pandas as pd

df = pd.DataFrame({
    "order_id": [1, 2, 3, 4, 5, 6],
    "city": ["Delhi", "Noida", "Delhi", "Pune", "Noida", "Delhi"],
    "category": ["A", "B", "A", "C", "B", "C"],
    "amount": [500, 250, 300, None, 900, 120],
    "date": ["2026-01-05", "2026-01-09", "2026-02-02", "2026-02-11", "2026-02-20", "2026-03-01"],
})
print(df.shape)
```

What is the difference between a Series and a DataFrame?

A Series is a one-dimensional labelled array, such as one column. A DataFrame is a two-dimensional table of Series that share an index.

How do I inspect a DataFrame?

`df.head()`, `df.info()` (types and non-null counts), `df.describe()` (summary statistics), `df.shape`, `df.dtypes` and `df.columns`. Always inspect before cleaning.

What is the difference between loc and iloc?

`loc` selects by label and includes both ends of a slice. `iloc` selects by integer position and excludes the end.

```
print(df.loc[0:2, "city"].tolist())
print(df.iloc[0:2, 1].tolist())
```

How do I filter rows?

Use boolean conditions with `&`, `|` and `~`, with each condition in parentheses.

```
print(df[(df["city"] == "Delhi") & (df["amount"] > 200)]["order_id"].tolist())
```

`df.query("city == 'Delhi' and amount > 200")` gives the same result in a readable form.

How do I handle missing values?

```
print(df["amount"].isna().sum())
filled = df["amount"].fillna(df["amount"].median())
print(filled.tolist())
clean = df.dropna(subset=["amount"])
print(len(clean))
```

Choose between dropping and filling by asking why values are missing, and record your choice.

How does groupby work?

`groupby` splits data into groups, applies a function and combines the results.

```
summary = df.groupby("city")["amount"].agg(["count", "sum", "mean"])
print(summary)
```

Use `.reset_index()` to turn the group keys back into columns, and `agg` with several functions or named outputs for more control.

What is the difference between merge, join and concat?

`merge` combines tables on key columns like a SQL join, with `how` set to inner, left, right or outer. `join` merges on the index. `concat` stacks tables vertically or side by side.

```
cities = pd.DataFrame({"city": ["Delhi", "Noida"], "state": ["Delhi", "UP"]})
m = df.merge(cities, on="city", how="left")
print(m[["order_id", "state"]].fillna("NA").values.tolist())
```

After a merge, compare the row count with what you expect to catch duplicated keys.

How do I work with dates?

```
df["date"] = pd.to_datetime(df["date"])
df["month"] = df["date"].dt.month
print(df.groupby("month")["order_id"].count().tolist())
```

Use `dt.year`, `dt.day_name()` and `resample` for time series.

How do I create a pivot table?

```
pt = df.pivot_table(index="city", columns="category", values="amount", aggfunc="sum", fill_value=0)
print(pt)
```

What are apply, map and vectorised operations?

Vectorised operations such as `df["amount"] * 2` run in optimised code and are fast. `map` transforms a Series with a dictionary or function. `apply` runs a function per row or column and is slower; use it only when no vectorised form exists.

How do I find and remove duplicates?

`df.duplicated()` flags repeats and `df.drop_duplicates(subset=["order_id"])` removes them. Decide which row to keep before dropping.

How do I speed up pandas?

Use vectorised operations, choose efficient types such as `category` for repeated text, read only needed columns, process large files in chunks and avoid loops over rows. Measure before optimising.

What is the SettingWithCopy warning?

It warns that you may be changing a copy of a slice instead of the original. Avoid chained indexing; use `.loc` to assign, or make an explicit copy with `.copy()`.

What practical tasks are common?

Typical tasks are: load a CSV and report missing values per column, find the top three categories by total amount, compute month-on-month change, merge two tables and find rows without a match, and reshape wide data to long with `melt`. Practise each on a small table so you can write the code quickly, explain every step and check the result with a second method such as a manual total.

How do I explain my pandas projects?

Say which dataset you used, how you cleaned it, which grouping or merging steps produced the key result, what chart you made and how you checked your numbers. Mention one bug you found, such as duplicated rows after a merge, and how you fixed it. A concrete example shows real hands-on use.

Frequently asked questions

What is the difference between loc and iloc in pandas?

loc selects by label and includes the end of a slice, while iloc selects by integer position and excludes the end.

How do I handle missing values in pandas?

Use isna to find them, then fillna to fill or dropna to remove, choosing based on why the values are missing.

What is the difference between merge and concat?

merge combines tables on key columns like a SQL join, while concat stacks tables vertically or side by side.

Why is apply slow?

apply runs a Python function on each row or column, while vectorised operations run in optimised compiled code.

What does the SettingWithCopyWarning mean?

You may be modifying a copy of a slice instead of the original DataFrame. Use loc for assignment or make an explicit copy.

Sources and further reading

- pandas documentation
- pandas user guide

About this PDF

These PDFs are study material for learning and revision. They are not a certificate, a degree or a promise of a job, and technology changes, so check official documentation for the latest details. Spot an error? Write to info@uncodemy.com and the trainer team will review it.

More free resources and courses: uncodemy.com | info@uncodemy.com | +91 98183 66550