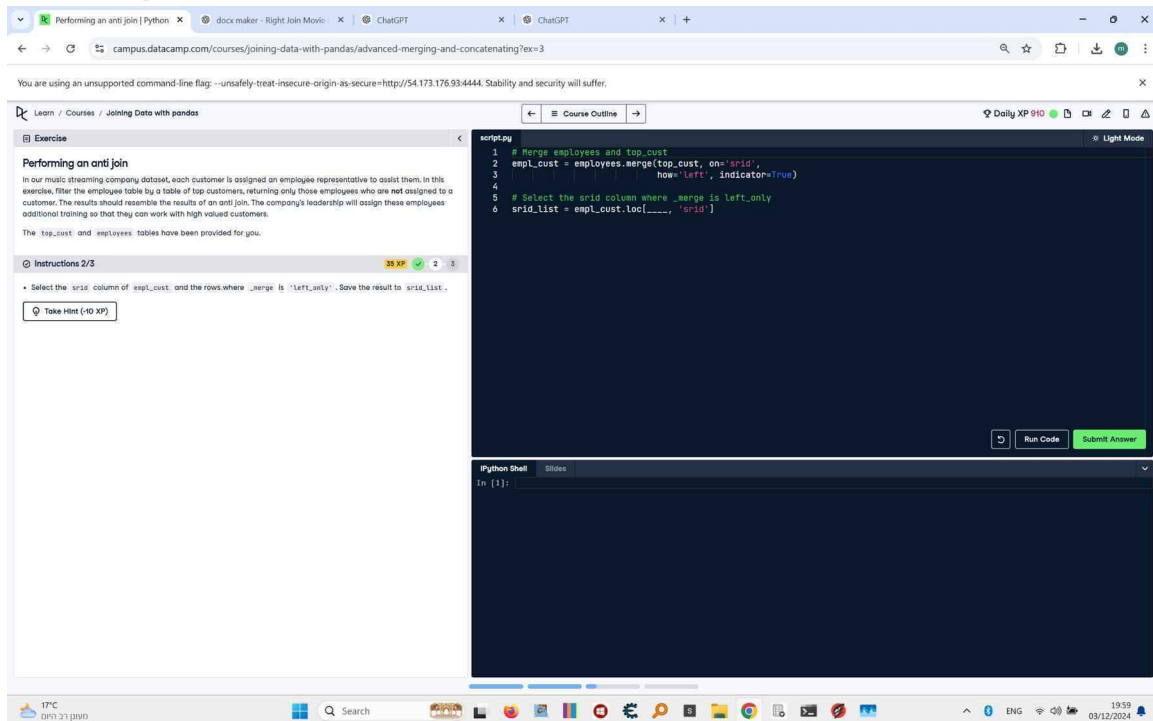


Filtering Rows in an Anti Join in Pandas



The screenshot shows a web browser window with a pandas exercise interface. The exercise is titled "Performing an anti join" and describes a task to filter employees based on a left join with top customers. The code editor shows the solution:

```
1 # Merge employees and top_cust
2 empl_cust = employees.merge(top_cust, on='srid', how='left', indicator=True)
3
4
5 # Select the srid column where _merge is left_only
6 srid_list = empl_cust.loc[_, '_merge' == 'left_only', 'srid']
```

Screenshot showing the exercise context for filtering rows in an anti join in pandas.

Code Answer:

```
# Merge employees and top_cust with a left join, setting indicator to True
empl_cust = employees.merge(top_cust, on='srid', how='left',
indicator=True)
```

```
# Select the srid column where _merge is left_only
srid_list = empl_cust.loc[empl_cust['_merge'] == 'left_only', 'srid']
```

Explanation:

1. The `merge` function combines the 'employees' table and 'top_cust' table based on the 'srid' column. The `how='left'` parameter ensures that all rows from the 'employees' table are included, and the `indicator=True` parameter creates a new column, `_merge`, indicating the source of each row.

2. The ``loc`` method is used to filter rows where the `'_merge'` column has the value `'left_only'`, which means these rows exist only in the `'employees'` table and not in the `'top_cust'` table.

3. The `'srid'` column from these filtered rows is selected, resulting in a list of `'srid'` values representing employees who are not assigned to any top customer.