

The screenshot shows a web browser window with the URL `projects.datacamp.com/projects/619`. The page is titled 'Project: The Android App Market on Google Play'. On the left, there are 'Task 7: Instructions' which include:

- Use a strip plot to visualize the distribution of paid apps across different categories.
- Plot a strip plot with x-axis extending along the `Price` range and y-axis depicting the `Category`.
- Find apps priced above \$200. Print the `Category`, `App` and `Price` columns for such apps.

On the right, a Jupyter notebook is open. It shows a strip plot of app prices by category. Below the plot, there is a code cell with the following code:

```
In [64]: import matplotlib.pyplot as plt
fig, ax = plt.subplots()
fig.set_size_inches(15, 8)

# Select a few popular app categories
popular_app_cats = apps[apps.Category.isin(['GAME', 'FAMILY', 'PHOTOGRAPHY',
'MEDICAL', 'TOOLS', 'FINANCE',
'LIFESTYLE', 'BUSINESS'])]

# Examine the price trend by plotting Price vs Category
ax = sns.stripplot(x=popular_app_cats..., y=popular_app_cats..., jitter=True, linewidth=1)
ax.set_title('App pricing trend across categories')

# Apps whose Price is greater than 200
apps_above_200 = ...
apps_above_200[['Category', 'App', 'Price']]
```

Google Play Store Analysis - Task 7

Task 7 Instructions

1. Plot a strip plot with the x-axis extending along the `Price` range and the y-axis depicting the `Category`.
2. Filter apps priced above \$200 and print the `Category`, `App`, and `Price` columns for such apps.
3. Use the `stripplot()` function from the `seaborn` library to create the plot.

Correct Code Implementation

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt
```

```
# Step 1: Filter apps by popular categories
```

```
popular_app_cats = apps[apps.Category.isin(['GAME', 'FAMILY',
'PHOTOGRAPHY',
'MEDICAL', 'TOOLS', 'FINANCE',
'LIFESTYLE', 'BUSINESS'])]
```

```
# Step 2: Examine price trends by category using strip plot
```

```
fig, ax = plt.subplots()
```

```
fig.set_size_inches(15, 8)
```

```
ax = sns.stripplot(x=popular_app_cats['Price'],
y=popular_app_cats['Category'],
                    jitter=True, linewidth=1)
ax.set_title('App Pricing Trend Across Categories')

# Step 3: Filter apps where Price > $200 and print specific columns
apps_above_200 = apps[apps['Price'] > 200][['Category', 'App', 'Price']]
print(apps_above_200)
```

Explanation of the Code

1. **Filter Popular Categories**:
 - Select apps from categories of interest using the `isin()` method on the `Category` column.
2. **Strip Plot**:
 - The `stripplot()` function visualizes the distribution of app prices across categories.
 - Adding `jitter=True` and `linewidth=1` improves the clarity of data points.
3. **Filter Expensive Apps**:
 - Use the condition `apps['Price'] > 200` to identify apps priced above \$200.
 - The `[['Category', 'App', 'Price']]` selects only the relevant columns for display.