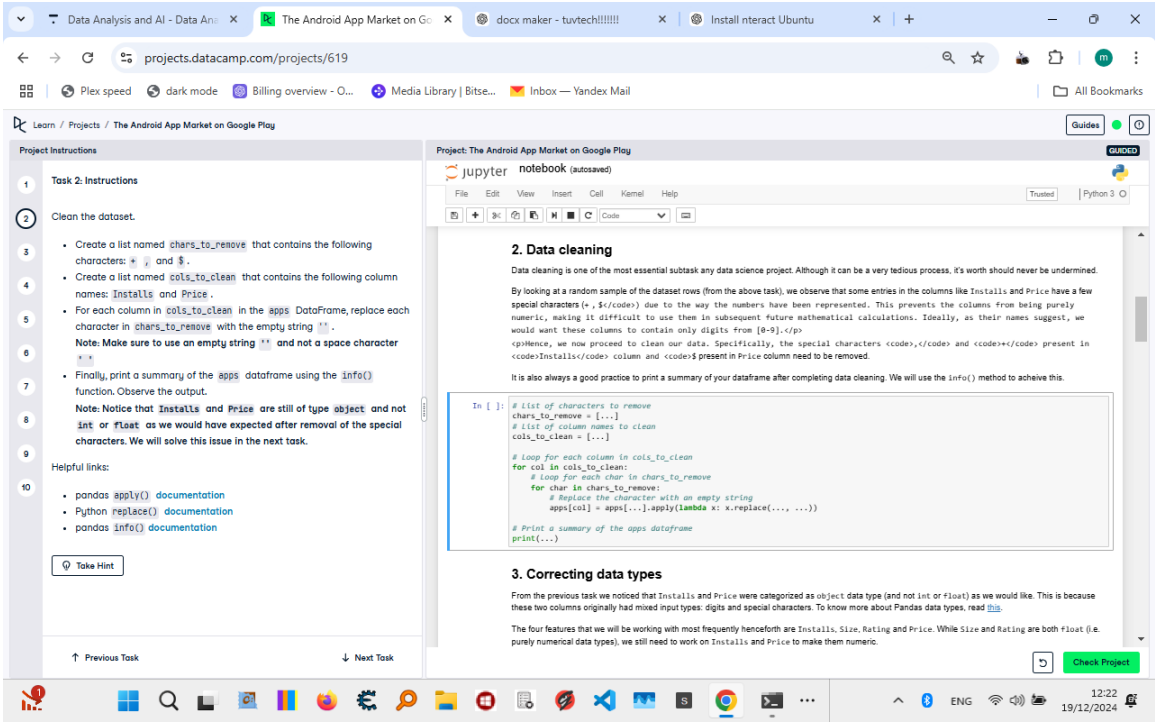

Google Play Store Analysis - Task 2

Task 2 Instructions

1. Clean the dataset:

   - Create a list named `chars_to_remove` that contains the following characters: `+` and `$`.
   - Create a list named `cols_to_clean` that contains the following column names: `Installs` and `Price`.
   - For each column in `cols_to_clean` in the `apps` DataFrame, replace each character in `chars_to_remove` with the empty string `''`.
   - Finally, print a summary of the `apps` DataFrame using the `info()` function.

Note: Make sure to use an empty string `''` and not a space character `' '`. Observe the output to ensure the columns are cleaned.

Correct Code Implementation

```
# Step 1: Create a list of characters to remove
chars_to_remove = ['+', '$']

# Step 2: Create a list of column names to clean
cols_to_clean = ['Installs', 'Price']
```

```
# Step 3: Loop over each column in 'cols_to_clean' and remove unwanted characters
for col in cols_to_clean:
    for char in chars_to_remove:
        apps[col] = apps[col].apply(lambda x: x.replace(char, '') if isinstance(x, str) else x)

# Step 4: Print a summary of the apps dataframe
print(apps.info())
```

Explanation of the Code

1. **chars_to_remove**: Specifies the special characters (`+` and `$`) to be removed from the specified columns.
2. **cols_to_clean**: Indicates the columns (`Installs` and `Price`) in which to perform the cleaning.
3. **Nested loop**:
   - Outer loop iterates over the columns in `cols_to_clean`.
   - Inner loop iterates over the characters in `chars_to_remove`, replacing each occurrence in the column with an empty string `''`.
4. **apply()**: Applies the `replace()` function to each element of the column, ensuring strings are cleaned.
5. **apps.info()**: Prints a summary of the DataFrame to verify that the cleaning operation has been performed.