

```
# Question 1: How many classes total?
total_classes = sum([len(course["courses"]) for course in course_dfs])
# returns 4902
```

The total classes can be found by taking the number of rows in each of the dataframes and adding them all together. This results in 4902 classes total.

```
# Question 2: Which department offers the most classes?
dept_most_classes = max([(len(course["courses"]),course.iloc[0,0][:4]) for course in course_dfs])[1]
# returns ECON
```

The department that offers the most classes can be found by taking the max of the number of rows of each of the course dataframes and then taking that dataframe's department. This results in the ECON department having the most classes.

```
# Question 3: Is there a difference in classes offered between quarters?
all_dfs = pd.concat(course_dfs)

def list_to_string(lst):
    '''turns list of strings into single string'''
    return "".join(lst)

all_dfs["terms_offered"] = all_dfs["terms_offered"].apply(list_to_string)
AUTUMN = "Autumn: " + str(len(all_dfs.query("'Autumn' in terms_offered")["courses"]))
WINTER = "Winter: " + str(len(all_dfs.query("'Winter' in terms_offered")["courses"]))
SPRING = "Spring: " + str(len(all_dfs.query("'Spring' in terms_offered")["courses"]))
SUMMER = "Summer: " + str(len(all_dfs.query("'Summer' in terms_offered")["courses"]))
print(", ".join([AUTUMN, WINTER, SPRING, SUMMER]))
# prints "Autumn: 1414, Winter: 1285, Spring: 1291, Summer: 58"
```

The apparent difference between quarters is shown here. By combining the dataframes together and then finding the rows of each quarter's newly called dataframe (using `df.query()`), we can find the number of classes offered in each quarter. Autumn is first with 1414 classes, then Spring with 1291, then Winter with 1285, and finally Summer with 58.