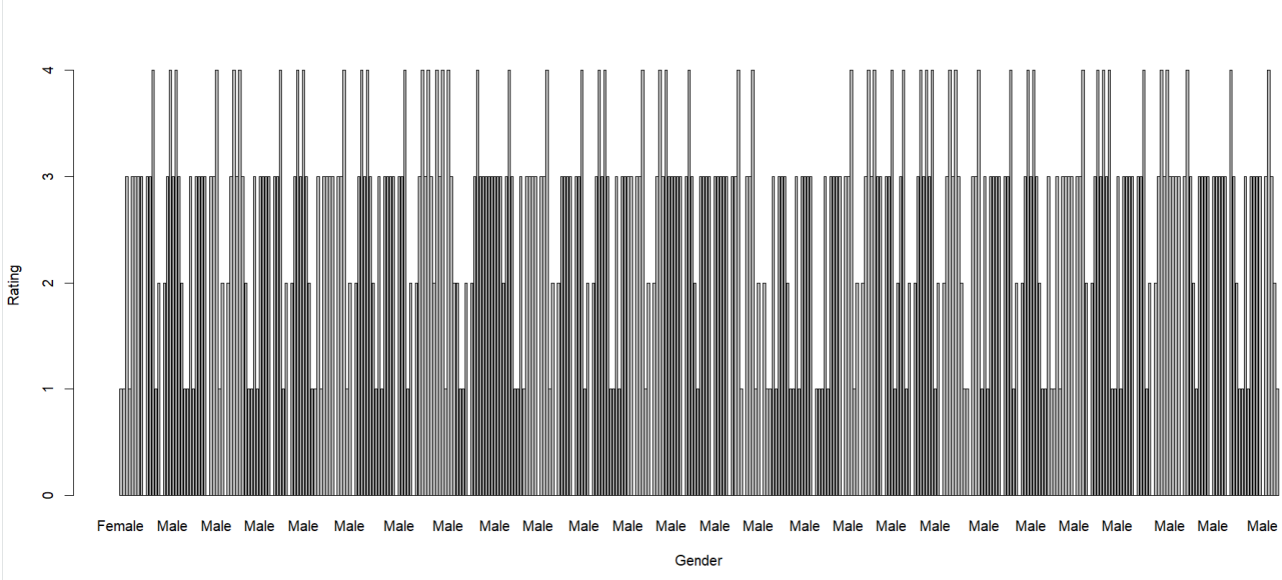
# DATA VISUALIZATION

Name :Vijayasharada Palakonda

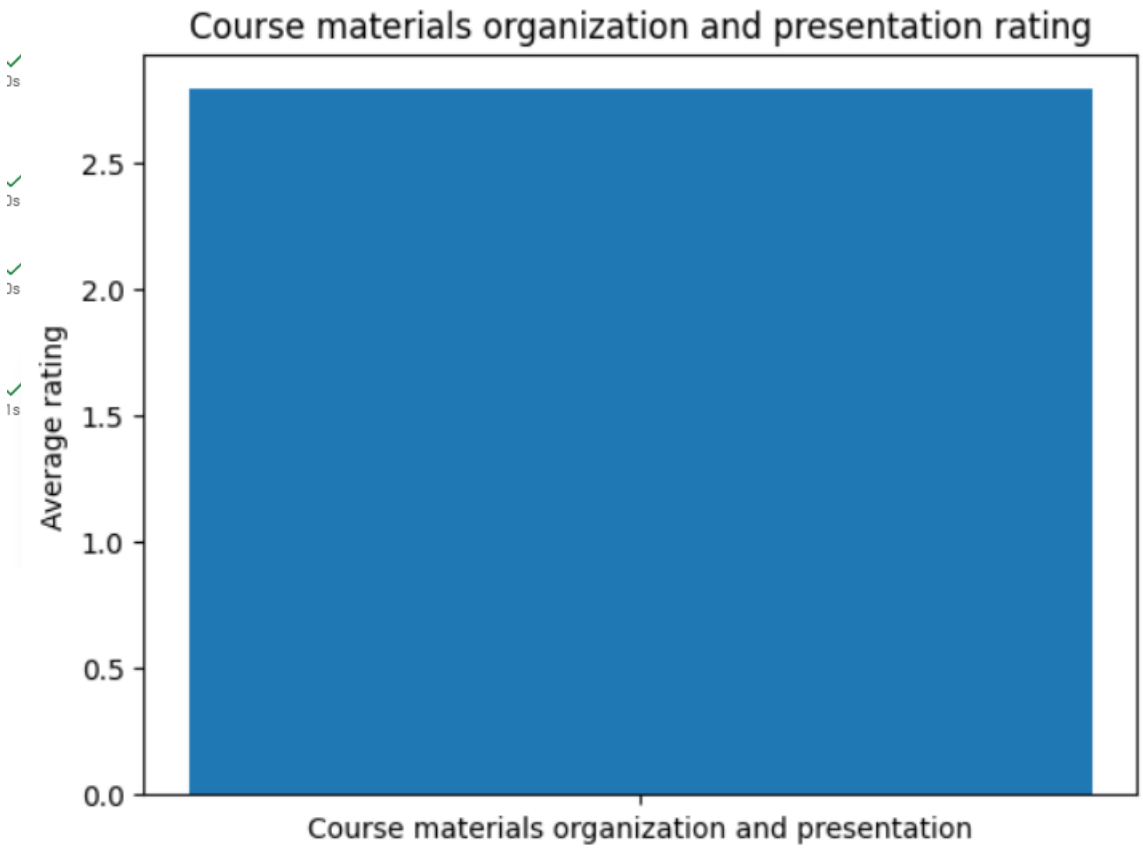
Reg no:20BCR7082

## 1.How would you rate the quality of the teaching and learning experience in

```
> df=read.csv("C:\\Users\\Vijaya Palakonda\\Desktop\\VITAP\\SEM6\\FDA\\LAB\\Untitled form (Responses) - Form Responses 1 (1).csv")
> barplot(df$what.would.you.rate.the..overall..course.on.the.basis.of.you.re.satisfaction)
> barplot(height=df$what.would.you.rate.the..overall..course.on.the.basis.of.you.re.satisfaction,names.arg=df$Gender,xlab="Gender",ylab="Rating")
>
```




1. How well were the course materials organized and presented?




2. How well did the course align with your career goals and interests?

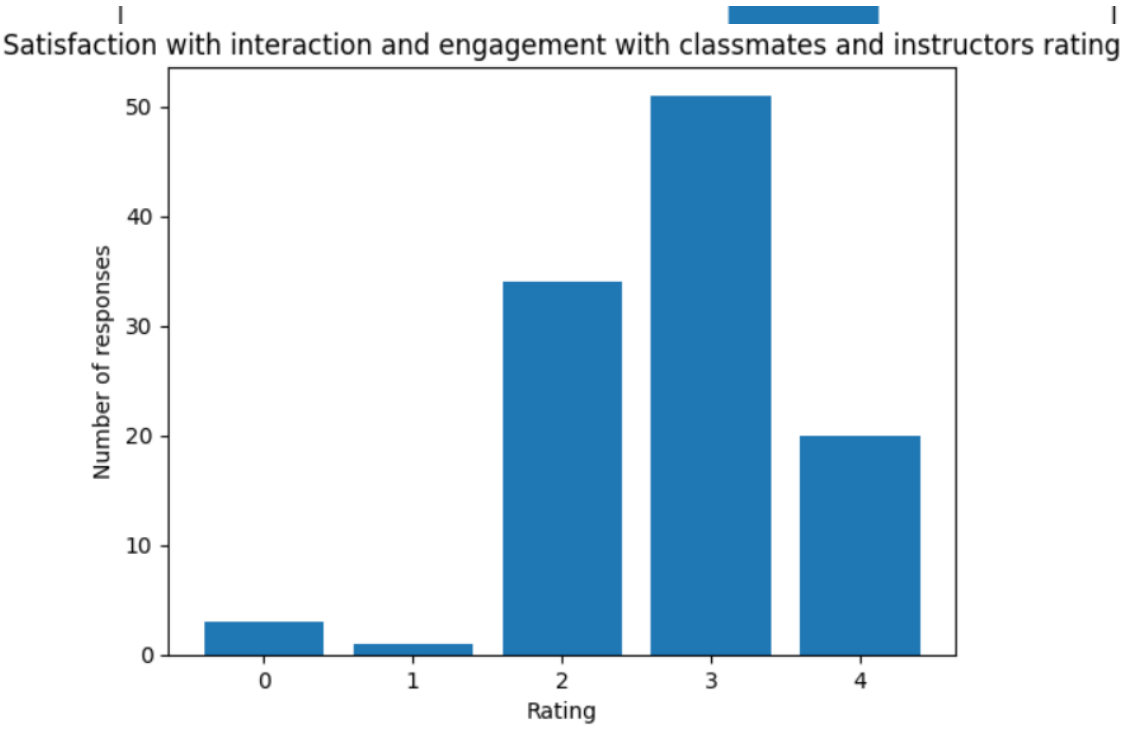
```
[5]  ratings_count = df["active_interest"].value_counts().sort_index()
```

```
fig, ax = plt.subplots()
ax.bar(ratings_count.index, ratings_count.values)
ax.set_xlabel("Rating")
ax.set_ylabel("Number of responses")
ax.set_title("Course alignment with career goals and interests rating")
plt.show()
```

3. 3How satisfied were you with the level of interaction and engagement with your classmates and instructors?

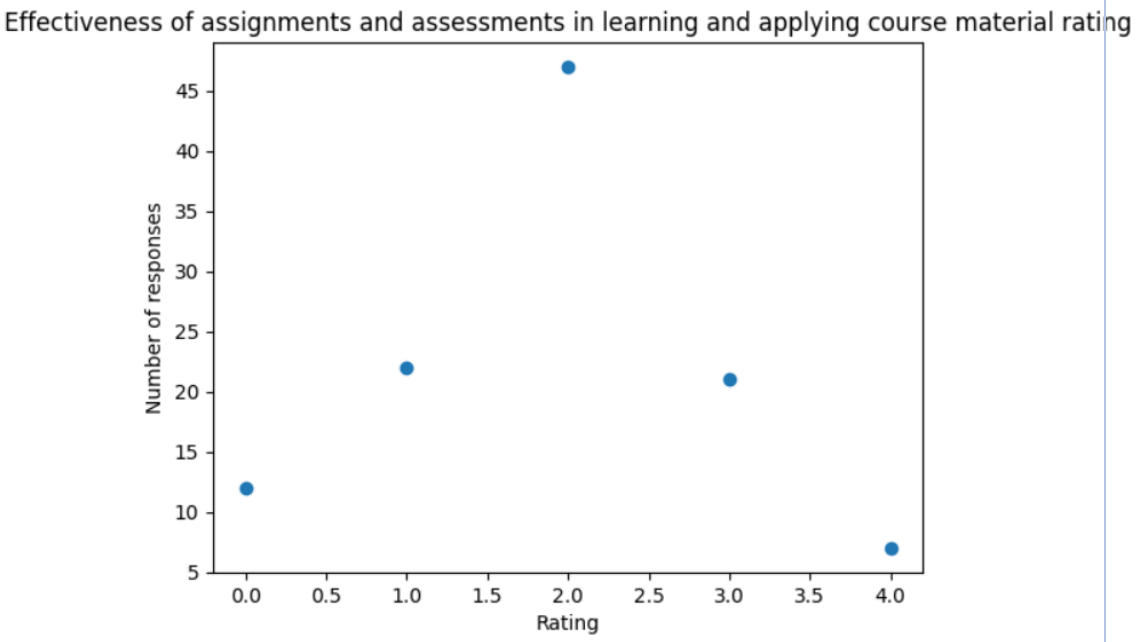
```
7]  ratings_count = df["teachers_able_communicate"].value_counts().sort_index()
```

```
fig, ax = plt.subplots()
ax.bar(ratings_count.index, ratings_count.values)
ax.set_xlabel("Rating")
ax.set_ylabel("Number of responses")
ax.set_title("Satisfaction with interaction and engagement with classmates and instructors rating")
plt.show()
```



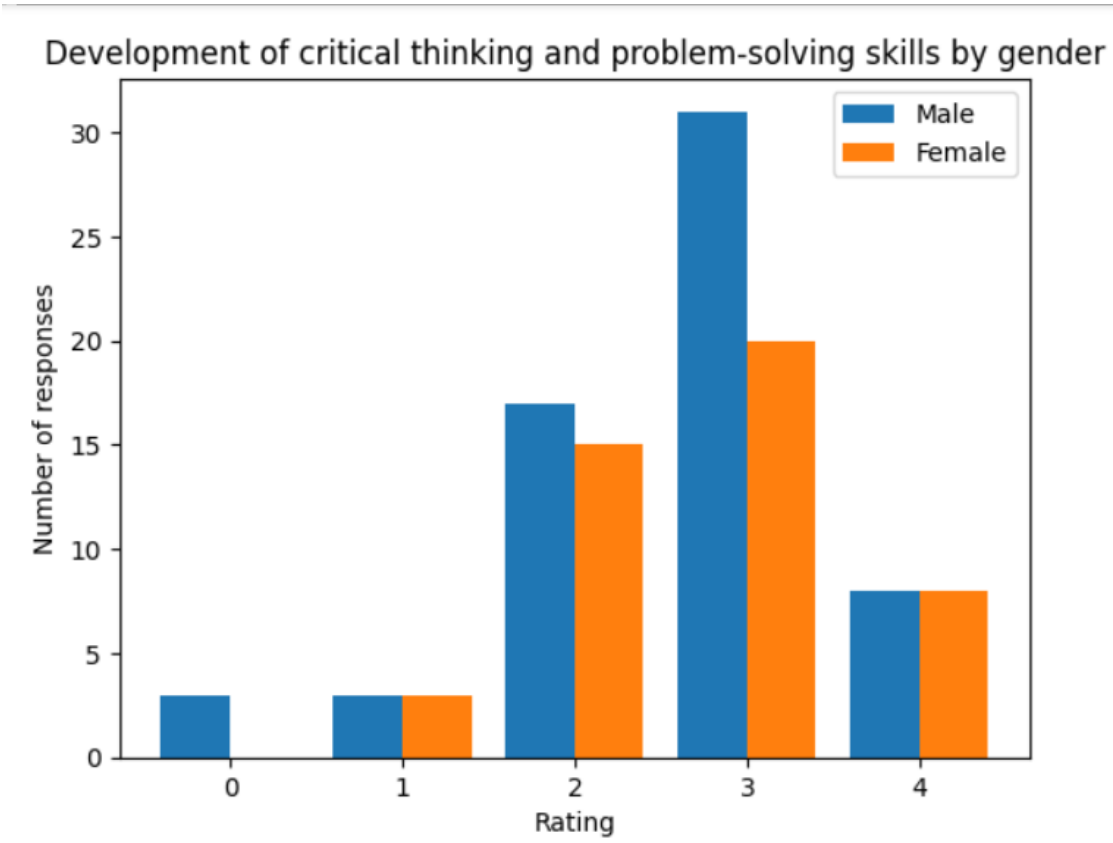

4. How effective were the assignments and assessments in helping you learn and apply the course material?

```
ratings_count = df["Teachers_encourage_you_to_participate_in_extracurricular_activities"].value_counts().sort_index()

fig, ax = plt.subplots()
ax.scatter(ratings_count.index, ratings_count.values)
ax.set_xlabel("Rating")
ax.set_ylabel("Number of responses")
ax.set_title("Effectiveness of assignments and assessments in learning and applying course material rating")
plt.show()
```



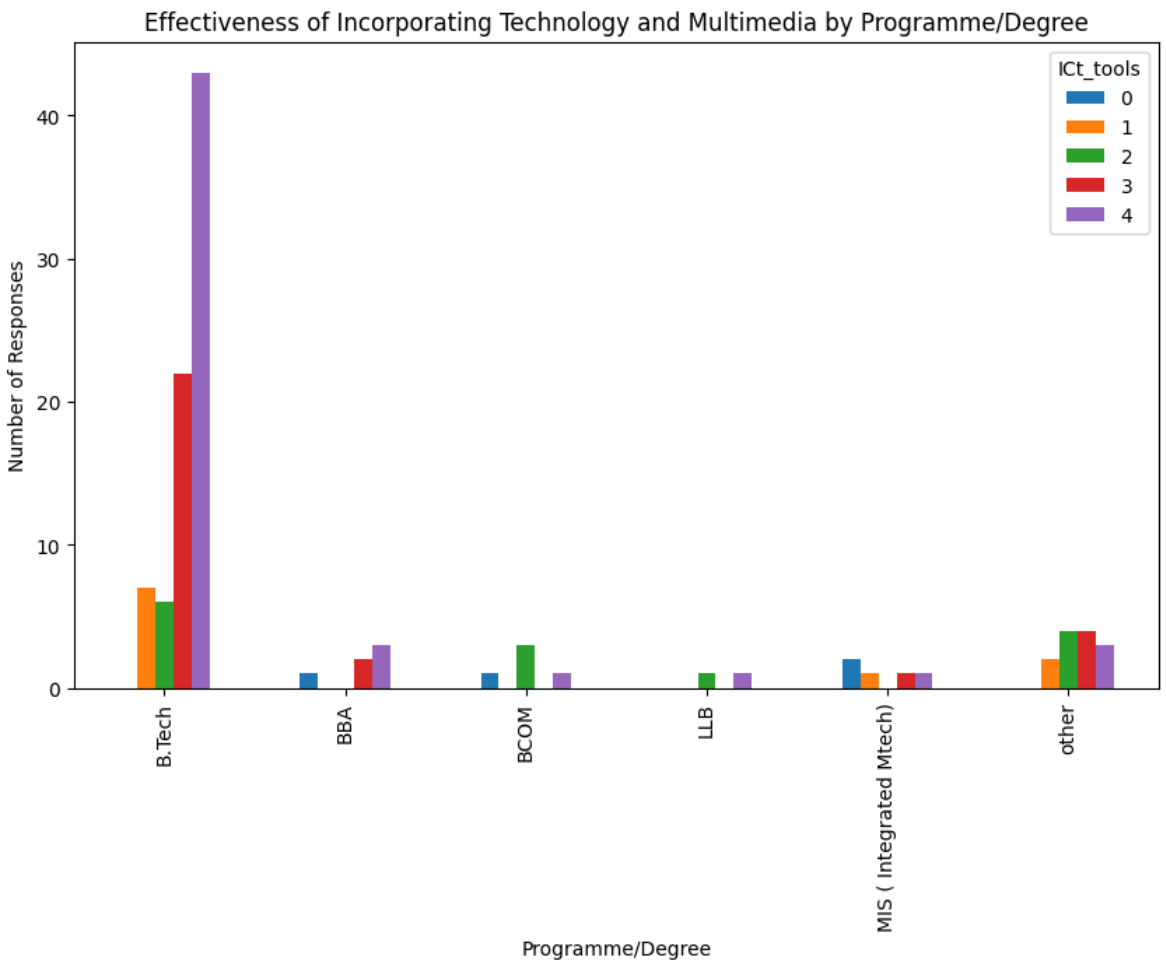

5. To what extent were you able to develop your critical thinking and problem-solving skills in this course?

```
target_question = "Efforts_are_made_by_the_teachers_to_inculcate_soft_skills_life_skills_and_employability _skills_to_make_you_ready _for_the_world_of_work
male_responses = df.loc[df["Gender"] == "Male", target_question]
female_responses = df.loc[df["Gender"] == "Female", target_question]

# Count the number of responses for each rating value (0-4)
male_ratings_count = male_responses.value_counts().sort_index()
female_ratings_count = female_responses.value_counts().sort_index()

# Plot the bar chart
fig, ax = plt.subplots()
ax.bar(male_ratings_count.index - 0.2, male_ratings_count.values, width=0.4, label="Male")
ax.bar(female_ratings_count.index + 0.2, female_ratings_count.values, width=0.4, label="Female")
ax.set_xlabel("Rating")
ax.set_ylabel("Number of responses")
ax.set_title("Development of critical thinking and problem-solving skills by gender")
ax.set_xticks(range(5))
ax.set_xticklabels(["0", "1", "2", "3", "4"])
ax.legend()
plt.show()
```




6. Were the instructors able to effectively incorporate technology and multimedia into their teaching?

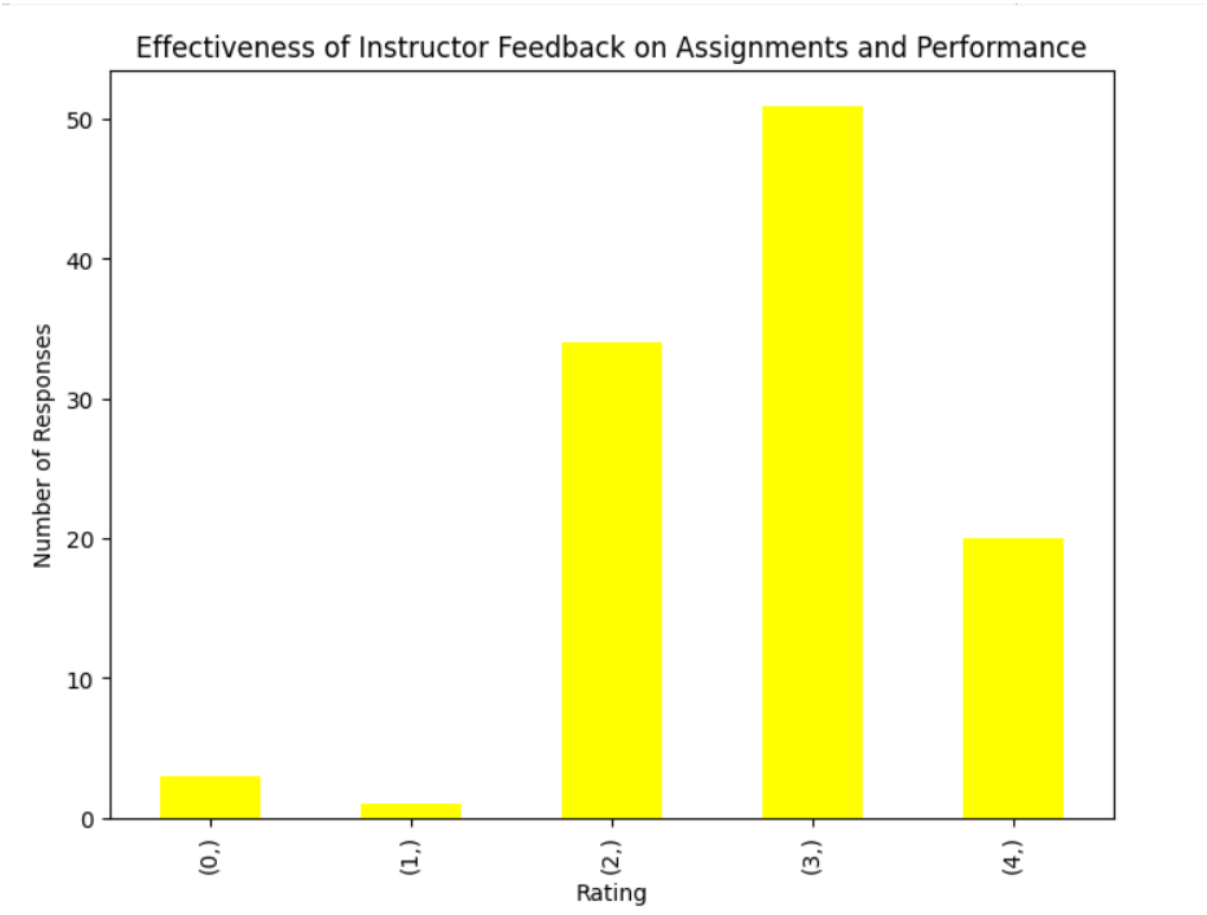
```
df = pd.read_csv('/content/Untitled form (Responses) - Form Responses 1 (1).csv')

# Filter the data for the specific question
df_q6 = df[['Programme/Degree', 'ICt_tools']]

# Group the data by programme and count the number of responses for each value
df_q6_grouped = df_q6.groupby(['Programme/Degree', 'ICt_tools']).size().unstack(fill_value=0)

# Create the bar plot
ax = df_q6_grouped.plot(kind='bar', figsize=(10,6))

# Set the axis labels and title
ax.set_xlabel('Programme/Degree')
ax.set_ylabel('Number of Responses')
ax.set_title('Effectiveness of Incorporating Technology and Multimedia by Programme/Degree')

# Show the plot
plt.show()
```




7. How well did the instructors provide feedback on your assignments and performance in the
course?

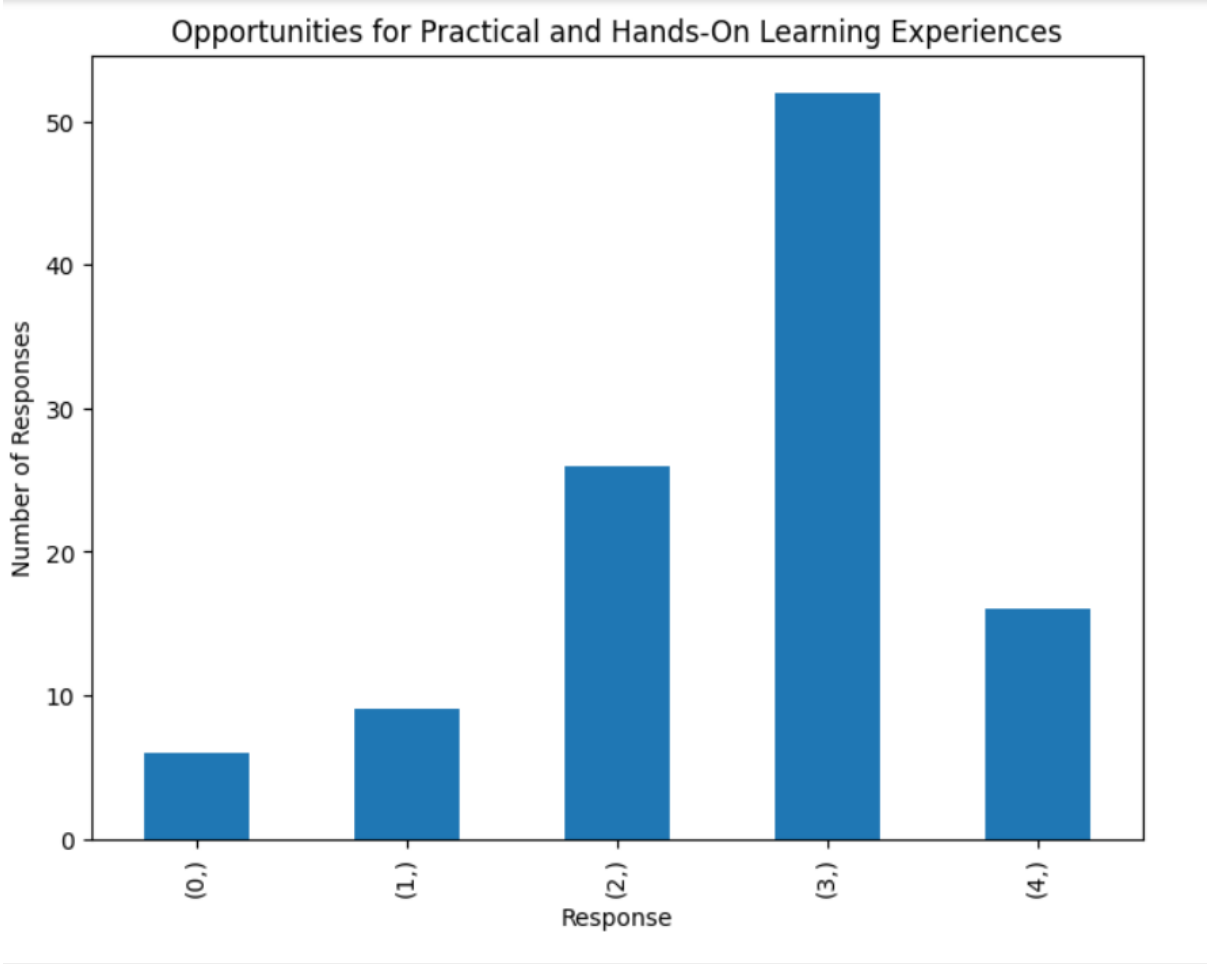
```
[3]  df = pd.read_csv('/content/Untitled form (Responses) - Form Responses 1 (1).csv')
```

```
df_q1 = df[['teachers_able_communicate']]

# Count the number of responses for each value
df_q1_counts = df_q1.value_counts(sort=False)

# Create the bar plot
ax = df_q1_counts.plot(kind='bar', color='yellow', figsize=(8,6))

# Set the axis labels and title
ax.set_xlabel('Rating')
ax.set_ylabel('Number of Responses')
ax.set_title('Effectiveness of Instructor Feedback on Assignments and Performance')

# Show the plot
plt.show()
```




8. Did the course provide opportunities for practical and hands-on learning experiences?

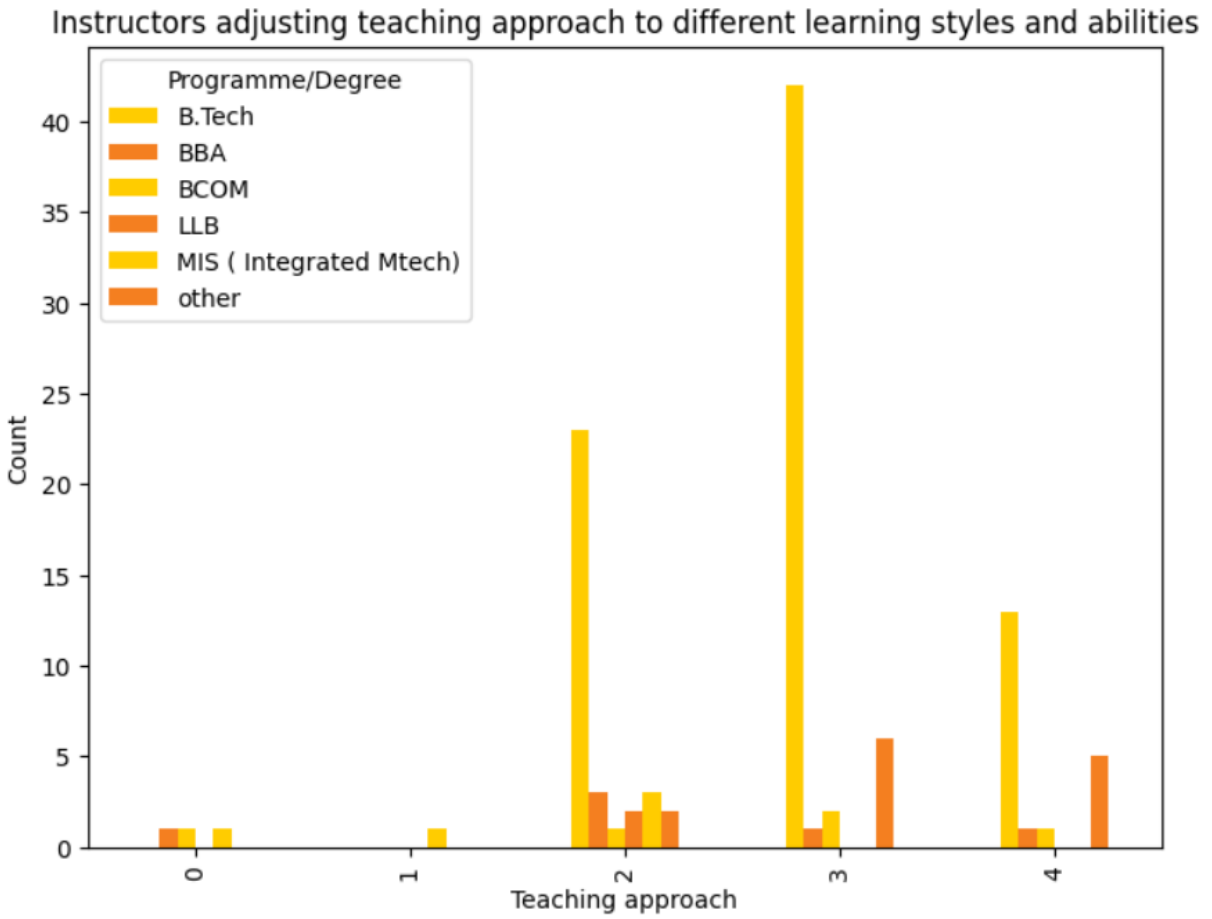
```
df_q9 = df[['The_teachers_illustrate_the_concepts_through_examples_and_applications']]

# Count the number of responses for each value
df_q9_counts = df_q9.value_counts(sort=False)

# Create the bar plot
ax = df_q9_counts.plot(kind='bar', figsize=(8,6))

# Set the axis labels and title
ax.set_xlabel('Response')
ax.set_ylabel('Number of Responses')
ax.set_title('Opportunities for Practical and Hands-On Learning Experiences')

# Show the plot
plt.show()
```




9.

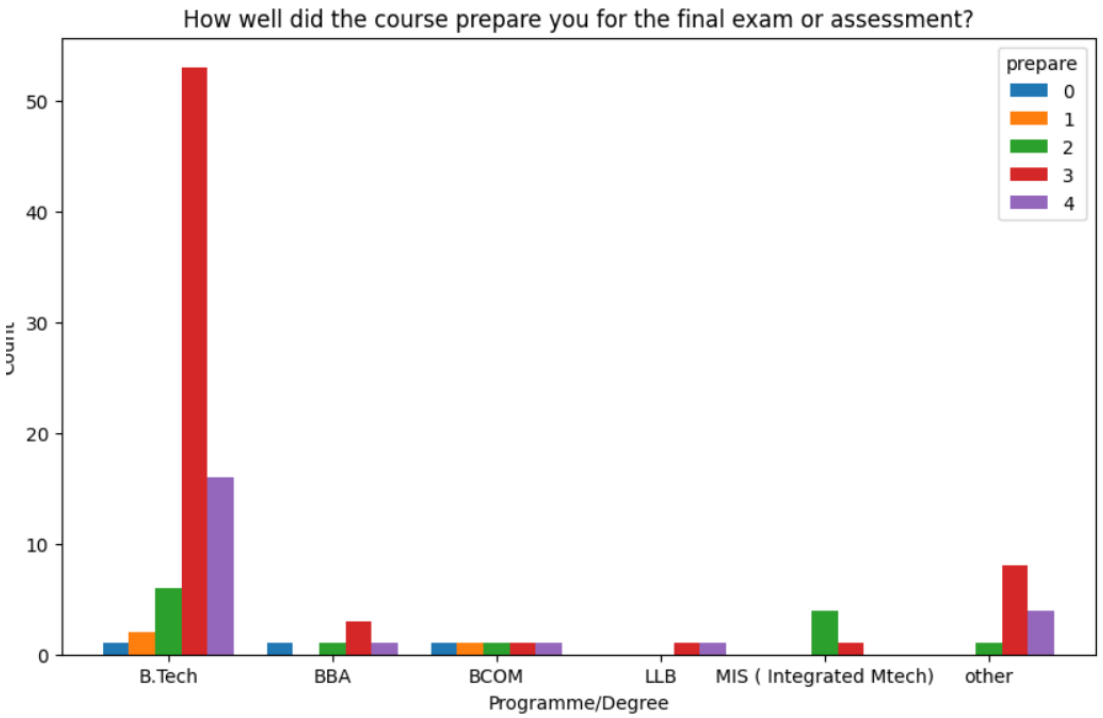
10.How well did the instructors adjust their teaching approach to accommodate different learning styles and abilities?

```
df_teaching = df[['Programme/Degree', 'teachers_able_communicate']]

# group by program and calculate value counts
df_teaching_counts = df_teaching.groupby(['Programme/Degree', 'teachers_able_communicate']).size().reset_index(name='count')

# pivot the data to have programs as columns
df_teaching_pivot = df_teaching_counts.pivot(index='teachers_able_communicate', columns='Programme/Degree', values='count')

# plot the bar chart
ax = df_teaching_pivot.plot(kind='bar', color=['#ffcc00', '#f47f20'], figsize=(8,6))
ax.set_title('Instructors adjusting teaching approach to different learning styles and abilities')
ax.set_xlabel('Teaching approach')
ax.set_ylabel('Count')
plt.show()
```



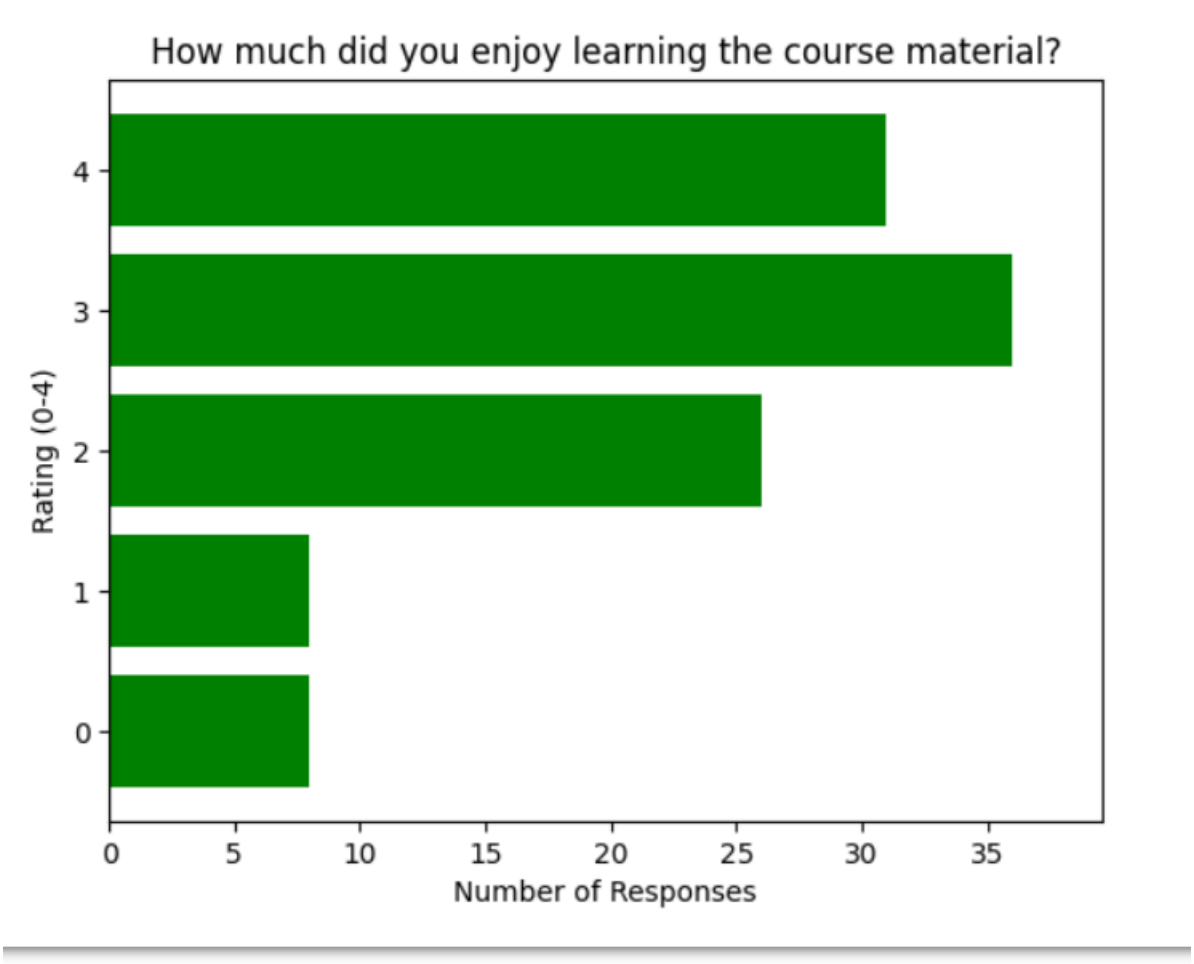

10.How well did the course prepare you for the final exam or assessment?

```
df = pd.read_csv('/content/Untitled form (Responses) - Form Responses 1 (1).csv')

# filter for the relevant columns
df_prep = df[['Programme/Degree', 'prepare']]

# group by program and calculate value counts
grouped_prep = df_prep.groupby('Programme/Degree')['prepare'].value_counts().unstack()

# plot the bar plot
ax = grouped_prep.plot(kind='bar', figsize=(10, 6), width=0.8)

# add labels and title
plt.title('How well did the course prepare you for the final exam or assessment?')
plt.xlabel('Programme/Degree')
plt.ylabel('Count')
plt.xticks(rotation=0)

# show the plot
plt.show()
```



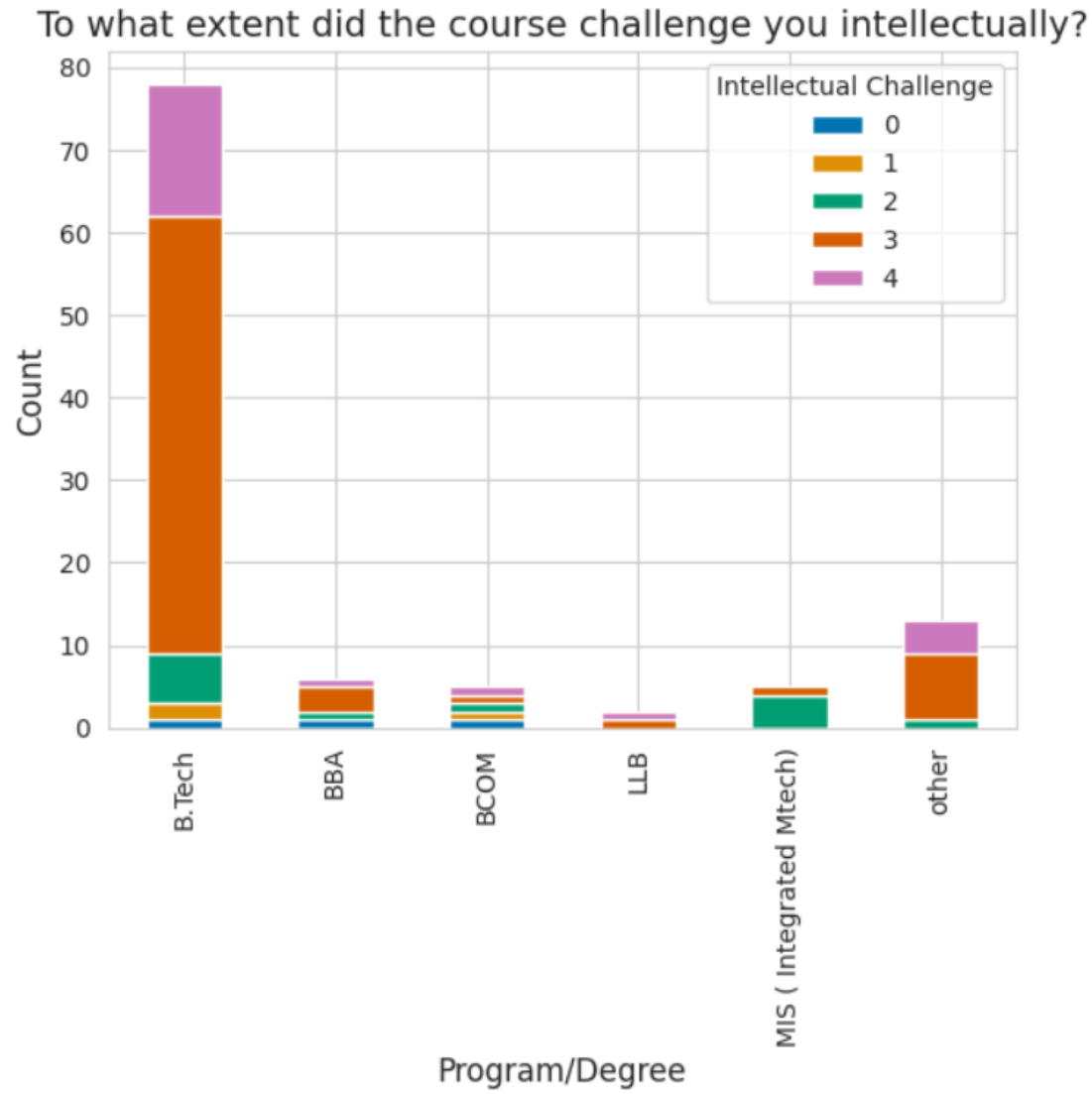

11.How much did you enjoy learning the course material?

```
df_enjoyment = df[['active_interest']]

# group by rating and calculate value counts
df_enjoyment_counts = df_enjoyment.groupby('active_interest').size().reset_index(name='counts')

# create the horizontal bar plot
plt.barh(df_enjoyment_counts['active_interest'], df_enjoyment_counts['counts'], color='green')
plt.title('How much did you enjoy learning the course material?')
plt.xlabel('Number of Responses')
plt.ylabel('Rating (0-4)')
plt.xlim(0, df_enjoyment_counts['counts'].max() * 1.1)
plt.show()
```




12.To what extent did the course challenge you intellectually?

```
counts = df.groupby(['Programme/Degree', 'prepare']).size().reset_index(name='count')

# pivot the table to make the response categories into columns
counts_pivot = counts.pivot(index='Programme/Degree', columns='prepare', values='count').fillna(0)

# create a stacked bar plot using Seaborn
sns.set_palette('colorblind')
ax = counts_pivot.plot(kind='bar', stacked=True)

# set the plot title and labels
ax.set_title("To what extent did the course challenge you intellectually?", fontsize=14)
ax.set_xlabel("Program/Degree", fontsize=12)
ax.set_ylabel("Count", fontsize=12)
ax.legend(title="Intellectual Challenge", bbox_to_anchor=(1, 1))

# display the plot
plt.show()
```

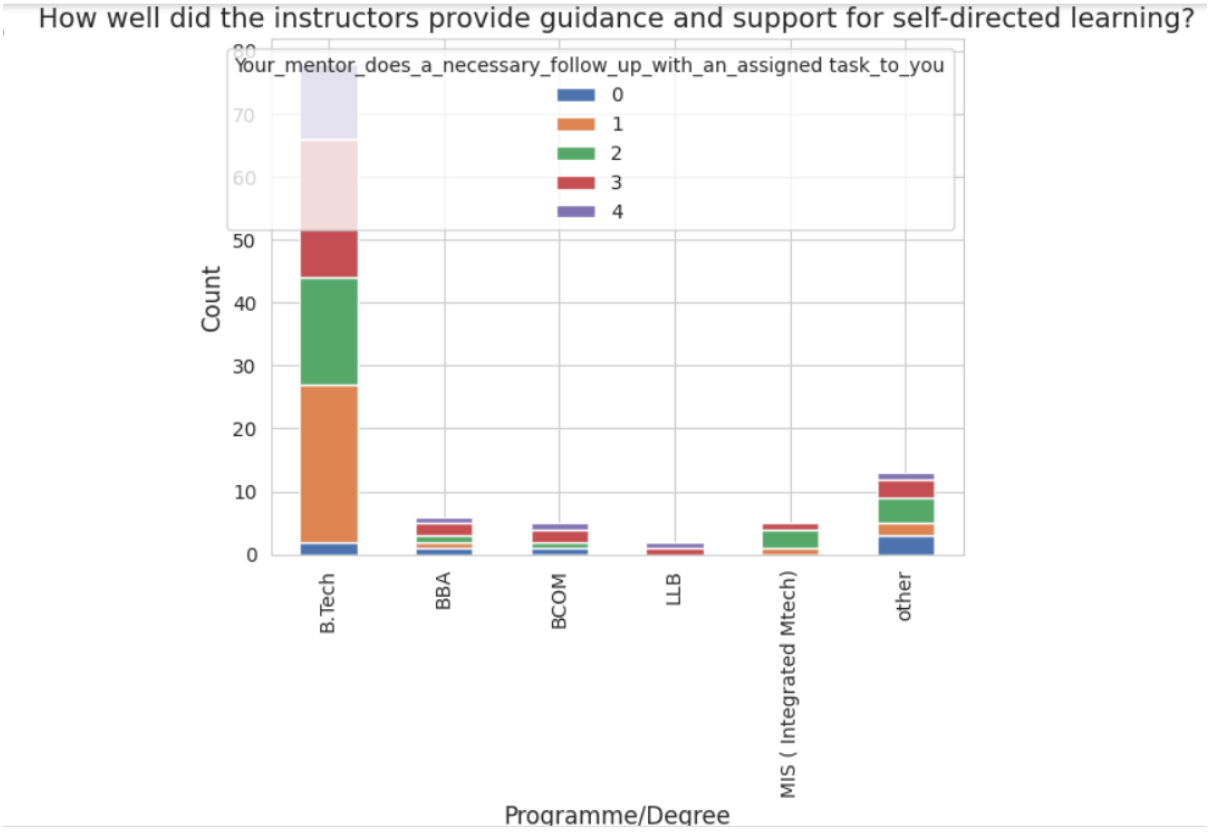



13. How well did the instructors provide guidance and support for self-directed learning?

```
import seaborn as sns
import matplotlib.pyplot as plt
```

```
df = pd.read_csv('/content/Untitled form (Responses) - Form Responses 1 (1).csv')
```

```
df_sdl = df[['Programme/Degree', 'Your_mentor_does_a_necessary_follow_up_with_an_assigned task_to_you']]

# group by program and calculate value counts
df_sdl_counts = df_sdl.groupby(['Programme/Degree', 'Your_mentor_does_a_necessary_follow_up_with_an_assigned task_to_you']).size().reset_index(r

# create a pivot table for easier plotting
df_sdl_pivot = df_sdl_counts.pivot(index='Programme/Degree', columns='Your_mentor_does_a_necessary_follow_up_with_an_assigned task_to_you', valu

# create a bar plot using Seaborn
sns.set_style("whitegrid")
sns.set_palette("deep")
ax = df_sdl_pivot.plot(kind='bar', stacked=True)

# set the plot title and labels
ax.set_title("How well did the instructors provide guidance and support for self-directed learning?", fontsize=14)
ax.set_xlabel("Programme/Degree", fontsize=12)
ax.set_ylabel("Count", fontsize=12)

# display the plot
plt.show()
```




14. How well did the instructors foster an inclusive and respectful learning environment?

```
df_incl_respect = df[['Programme/Degree', 'quality ']]

# group by program and calculate value counts
df_incl_respect_counts = df_incl_respect.groupby(['Programme/Degree', 'quality ']).size().reset_index(name='count')

# create a pivot table for easy plotting
df_incl_respect_pivot = df_incl_respect_counts.pivot(index='Programme/Degree', columns='quality ', values='count')

# create a bar plot using Seaborn
sns.set_style("whitegrid")
sns.set_palette("deep")
ax = df_incl_respect_pivot.plot(kind='bar', stacked=True, figsize=(8,6))

# set the plot title and labels
ax.set_title("How well did the instructors foster an inclusive and respectful learning environment?", fontsize=14)
ax.set_xlabel("Program/Degree", fontsize=12)
ax.set_ylabel("Count", fontsize=12)

# display the plot
plt.show()
```