FIT5145 - Introduction to Data Science

Assignment 1

The aim of this assignment is to investigate and visualise data using various data science tools. It will test your ability to:

1. read data files in Python and extract related data from those files;
2. wrangle and process data;
3. use various graphical and non-graphical tools to perform exploratory data analysis and visualisation;
4. use basic tools for managing and processing big data; and
5. communicate your findings in your report.

You will need to submit two files:

1. The Python code as a Jupyter notebook file that you wrote to analyse and plot the data.
2. A PDF of your Jupyter notebook file containing your answers (code, figures and answers to all the questions). Make sure to include screenshots/images of the graphs you generate in order to justify your answers to all the questions. Marks will be assigned to PDF reports based on their correctness and clarity. -- For example, higher marks will be given to PDF reports containing graphs with appropriately labelled axes.

IMPORTANT NOTE - Zip file submission will have a penalty of 10%. Do not submit the separate files requested above together in one Zip file. As indicated in the rubric, marks will be deducted for this because it adds significantly to the time it takes for the markers to open up and access your assignments given that there are many students in this class.

Tasks

There are two tasks that you need to complete for this assignment, Task A and Task B. You need to use Python to complete the tasks.

Task A - Who are Data Scientists? Data Scientist Demographics

'What does a does a Data Scientist look like?', *'What is Data Science exactly?'*, *'Is Python or R better to learn for beginners?'*, *'Do you have to have a degree in Computer Science to be a Data Scientist?'* and *'Do data scientists earn as much as I think?'*.

Anjul Bhambri, the Vice Preseident of big data prodcts at IBM says this

> *'A data scientist is somebody who is inquisitive, who can stare at data and spot trends. It's almost like a Renaissance individual who really wants to learn and bring change to an organisation.'*

In this course, you have learned that the diversity in definitions, skill sets, tools, applications and knowledge domains that make data science challenging to define precisely. By completing the following questions, we hope you can get a more precise understanding.

The Data

Kaggle is the home of analytics and predictive modelling competitions. Data Science enthusiast, beginners to professionals, compete to create the best predictive models using datasets uploaded both by individuals and companies looking for insights. Prizes can be as high as $3 million US. In late 2017 a survey of Kaggle users was conducted and received over 16,700 responses. The dataset was, of course, made public and many insights have emerged since. We have taken a portion of the data set and heavily modified the data. Both to clean the data, a significant component of data science and to ensure original assignment submission.

Your Job

The following notebook has been constructed to provide you with directions (blue), assessed questions (brown) and background information. Responses to both blue directions and brown questions are assessable.

You will be required to write your own code. Underneath direction boxes, there will be empty cells with the comment #Your code. Insert new cells under this cell if required.

To respond to questions you should double click on the cell beneath each question with the comment Answer.

Please note, your commenting and adherence to Python code standards will be marked. This notebook has been designed to give you a template for how we expect Python Notebooks to be submitted for assessment. If you require further information on Python standards, please visit https://www.python.org/dev/peps/pep-0008/ (https://www.python.org/dev/peps/pep-0008/) Do not change any of the directions or answer boxes, the order of questions, order of code entry cells or the name of the input files.

The Files

- *multipleChoiceResponses.csv * : Participants' answers to multiple choice questions. Each column contains the answers of one respondent to a specific question.
- *conversionRates.csv* : Currency conversion rates to USD.

** Your Information** Enter your information in the following cell. Please make sure you specify what version of python you are using as your tutor may not be using the same version and will adjust your code accordingly.

Student Information

Please enter your details here.

Name: Anik Dey Sarker

Student number: 29339472

Tutorial Day and Time: Wednesday,02:00 PM

Tutor: Dilini Rajapaksha Hewa Ranasinghage

Environment: Python 3.7.1 and Anaconda 4.5.12 (64-bit)

Table of contents

0. Load your libraries and files

1. ** Load your libraries and files**
This assesment will be conducted using pandas. You will also be required to create visualisations. We recomend Seaborn which is more visually appealing than matplotlib. However, you may choose either. For further information on Seaborn visit https://seaborn.pydata.org/ (https://seaborn.pydata.org/)

Hint: Remember to comment what each library does.

In [66]:

```
# Your code
import numpy as np       #performs scientific computing
import pandas as pd  #dataframe library, easy-to-use data structures and data analysis tool
import seaborn as sns   #It provides high-level interface for drawing informative statistic
import matplotlib.pyplot as plt #It provides an object-oriented API for putting plots into

%matplotlib inline
sns.set(rc={'figure.figsize':(12,8)})

multipleChoiceResponses=pd.read_csv('multipleChoiceResponses.csv')
conversionRates=pd.read_csv('conversionRates.csv')
```

1. Demographic Analysis

So what does a data scientist look like?

Let's get a general understanding of the characteristics of the survey participants. Demographic overviews are a standard way to start an exploration of survey data. The types of participants can heavily affect the survey responses.

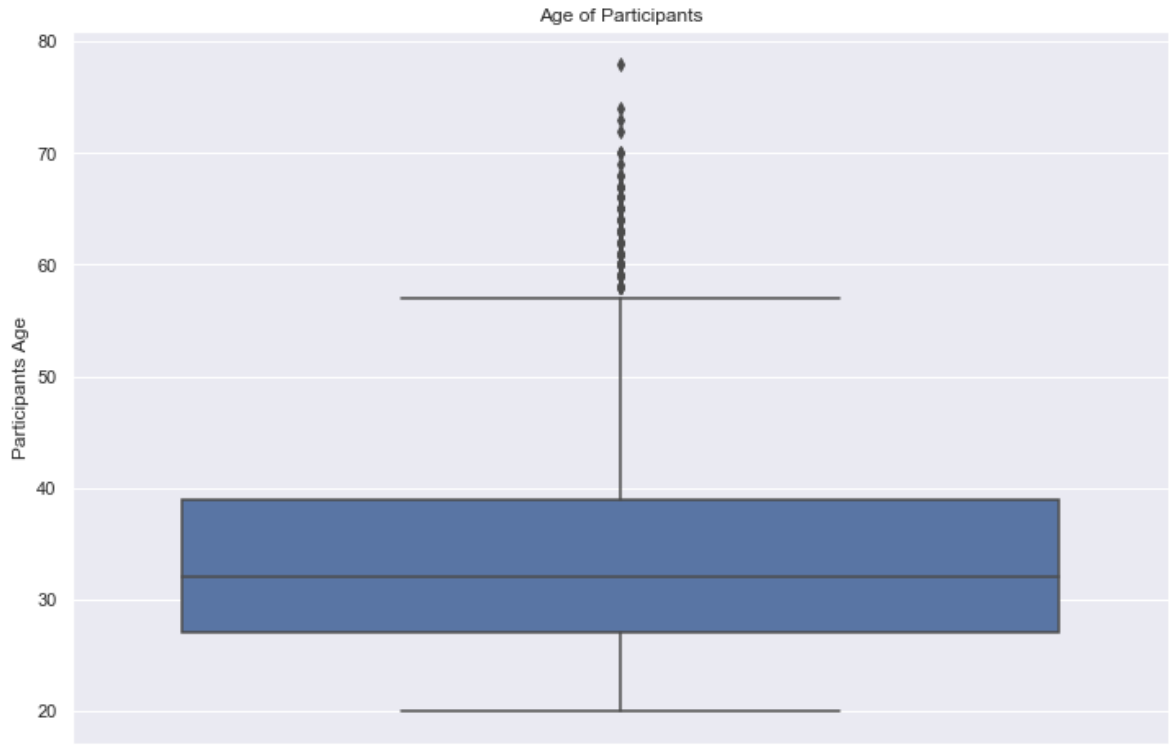
1.1 Age

Visualisation is a quick and easy way to gain an overview of the data. One method is through a boxplot. Boxplots are a way to show the distribution of numerical data and display the five descriptive statistics: minimum, first quartile, median, third quartile, and maximum. Outliers should also be shown.

2 Create a box plot showing the age of all the participants.
Your plot must have labels for each axis, a title, numerical points for the age axis and also show the outliers.

In [67]:

```
# Your code
sns.boxplot(y=multipleChoiceResponses['Age']).set_title('Age of Participants')
plt.ylabel('Participants Age')
plt.show()
```



3. Calculate the five descriptive statistics as shown on the boxplot, as well as the mean
Round your answer to the nearest whole number.

In [68]:

```
# Your code
mean = int(multipleChoiceResponses['Age'].mean())
median = int(multipleChoiceResponses['Age'].quantile(0.5))
q1 = int(multipleChoiceResponses['Age'].quantile(0.25))
q3 = int(multipleChoiceResponses['Age'].quantile(0.75))
iqr = q3 - q1
minimum = int(multipleChoiceResponses['Age'].min())
maximum = q3 + 1.5*iqr

print('Mean: ' + str(mean))
print('Minimum: ' + str(minimum))
print('First Quartile: ' + str(q1))
print('Median: ' + str(median))
print('Third Quartile: ' + str(q3))
print('Maximum: ' + str(maximum))
print('Outliers: ' + ','.join([str(age) for age in list(multipleChoiceResponses[multipleCho
```

```
Mean: 34
Minimum: 20
First Quartile: 27
Median: 32
Third Quartile: 39
Maximum: 57.0
Outliers: 61,63,73,66,60,67,62,70,59,74,64,58,78,68,65,69,72
```

Answer

Mean: 34
Minimum: 20
First Quartile: 27
Median: 32
Third Quartile: 39
Maximum: 57.0
Outliers: 61, 63, 73, 66, 60, 67, 62, 70, 59, 74, 64, 58, 78, 68, 65, 69, 72

4. Looking at the boxplot what general conclusion can you make about the age of the participants?
You must explain your answer concerning the median, minimum and maximum age of the respondents.
You must also make mention of the outliers if there are any.

Answer Almost half of the participants are aged between 27 and 39 years old. The youngest participant is 20 years old, and the eldest is 57 years old. The median of the age distribution is 32. There are some outliers aged between 61 and 72 inclusive.

5. Regardless of the errors that the data show, we are interested in working-age data scientists, aged between 18 and 65.
How many respondents were under 18 or over 65?

In [69]:

```
# Your code
multipleChoiceResponses[(multipleChoiceResponses['Age'] < 18) | (multipleChoiceResponses['A
```

Out[69]:

```
(19, 11)
```

Answer

19 respondents were under 18 or over 65.

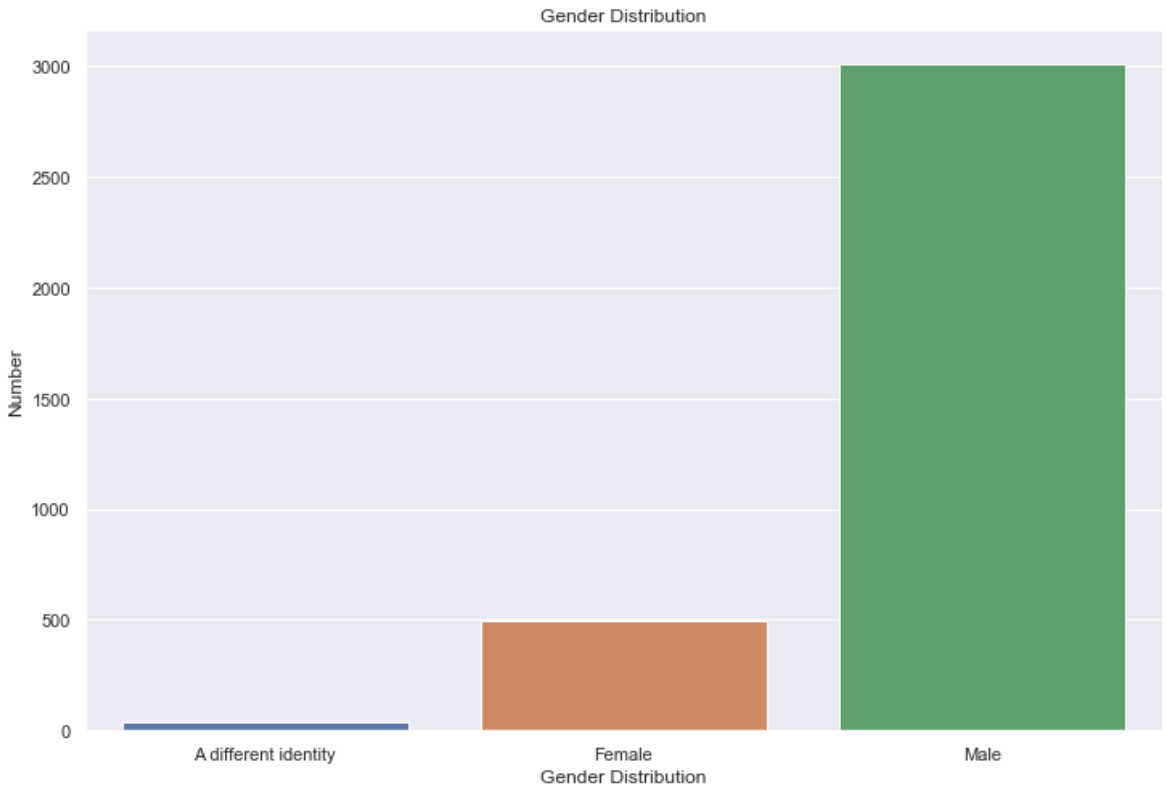
1.2 Gender

We are interested in the gender of respondents. Within the STEM fields, there are more males than females or other genders. In 2016 the Office of the chief scientist found that women held only 25% of jobs in STEM. Let's see how data science compares.

6. Plot the gender distribution of survey participants.

In [70]:

```
# Your code
gender = multipleChoiceResponses.groupby('GenderSelect').count()
sns.barplot(x=gender.index, y='CurrentJobTitleSelect', data=gender).set_title('Gender Distr
plt.xlabel('Gender Distribution')
plt.ylabel('Number')
plt.show()
```



7. What percentage of respondants were men? What percentage of respondants were women?

In [71]:

```
# Your code
gender = multipleChoiceResponses.groupby(['GenderSelect'])
Gender = gender[['CurrentJobTitleSelect']].count()
Gender.rename(columns={'CurrentJobTitleSelect':'percentage'},inplace=True)
Percentage=Gender.apply(lambda x:100* x/x.sum()).reset_index()
round(Percentage,2)
```

Out[71]:

	GenderSelect	percentage
0	A different identity	1.02
1	Female	14.01
2	Male	84.97

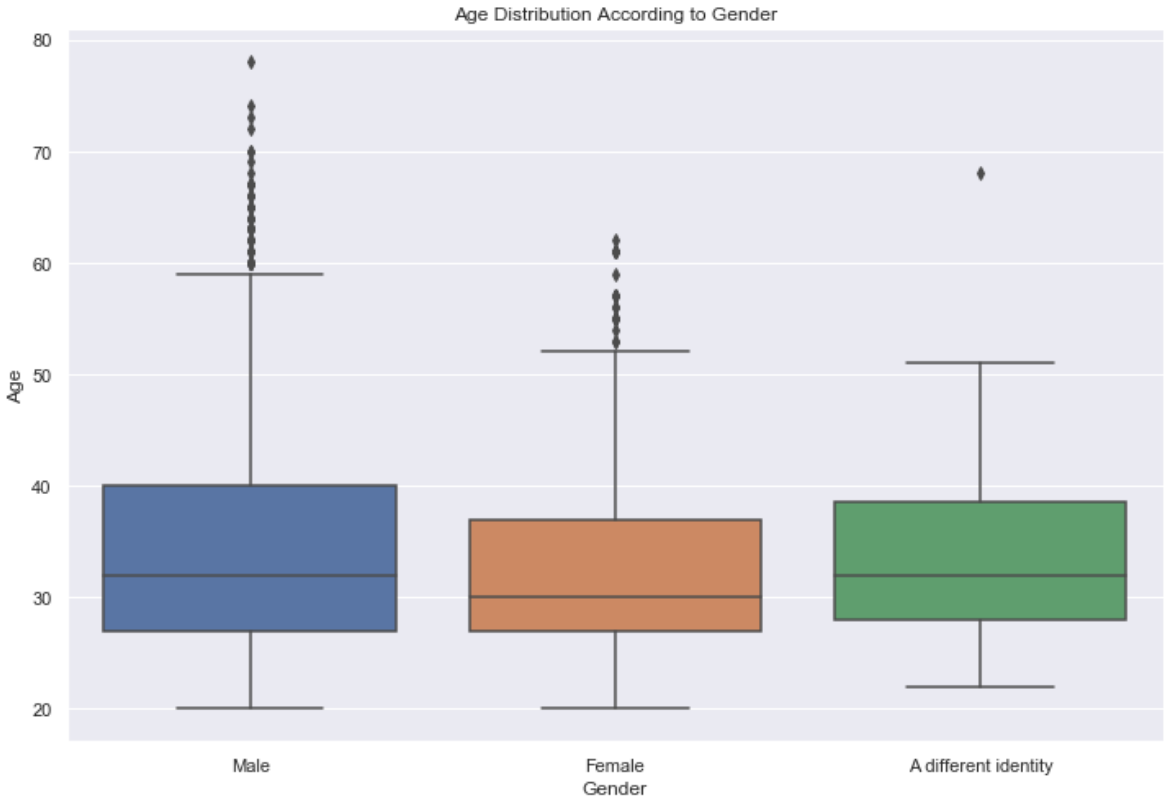
Answer 84.97% were Men,14.01% were Female and 1.02% were from a different identity

8. Let's see if there is any relationship between age and gender.
Create a box plot showing the age of all the participants according to gender.

Include the response 'Different identity' in your plot.

In [72]:

```
# Your code
sns.boxplot(x="GenderSelect", y="Age", data=multipleChoiceResponses).set_title('Age Distrib
plt.xlabel('Gender')
plt.show()
```



9. What comments can you make about the relationship between the age and gender of the respondents?

Hint: You need to determine the numeric descriptive statistics

In [73]:

```
# Your code
round(multipleChoiceResponses.groupby('GenderSelect')['Age'].agg('describe'),2)
```

Out[73]:

GenderSelect	count	mean	std	min	25%	50%	75%	max
A different identity	36.0	34.67	9.84	22.0	28.0	32.0	38.5	68.0
Female	496.0	32.74	8.66	20.0	27.0	30.0	37.0	62.0
Male	3008.0	34.64	9.56	20.0	27.0	32.0	40.0	78.0

Answer

The male respondants were relatively older than female respondants. Also, the male working age tends to be longer than their female counterparts.

1.3 Country

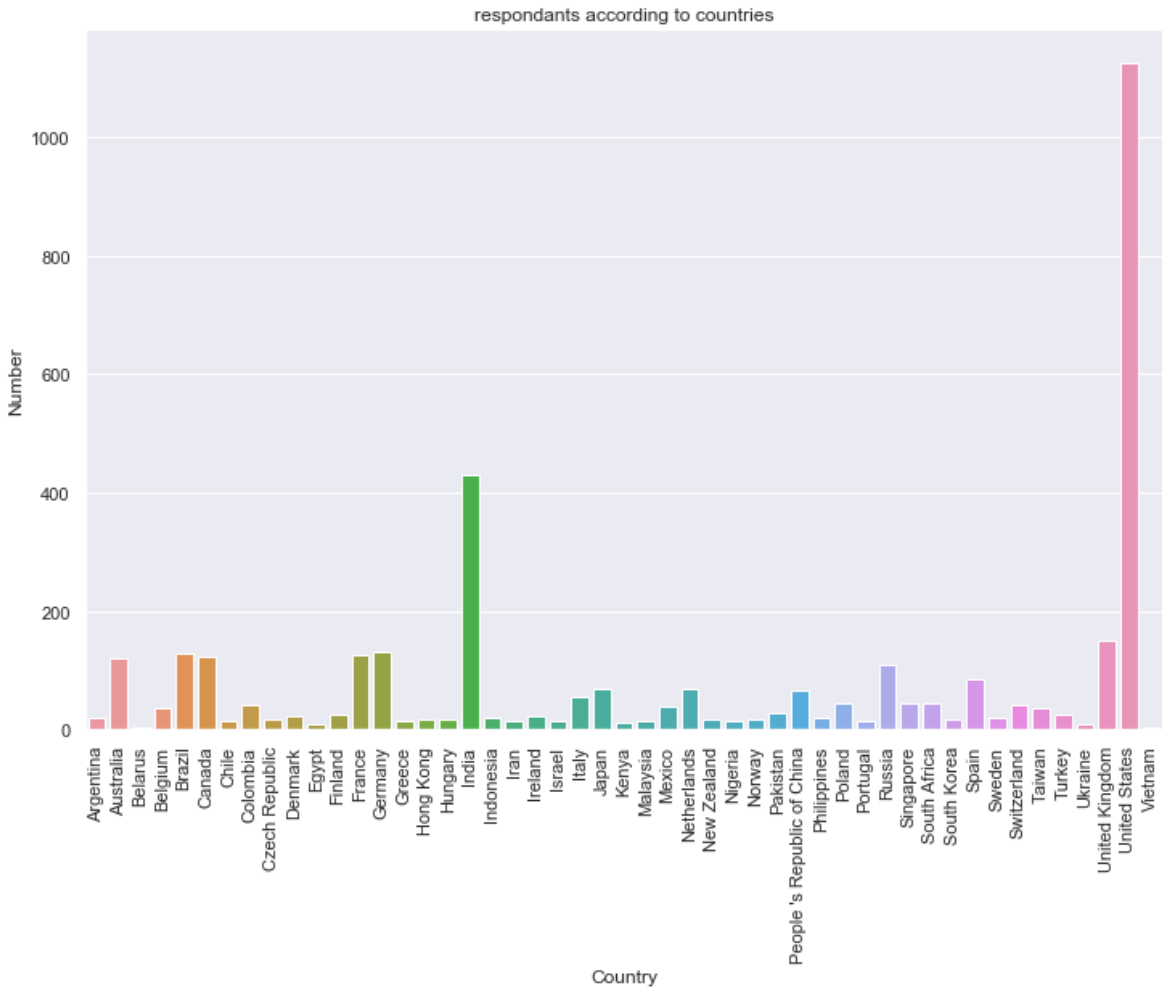
We know that people practise data science all over the world. The United States is thought of as a 'hub' of commercial data science as well as research followed by the United Kingdom and Germany.

Because the field is evolving so quickly, it may be that these perceptions, formed in the late 2000s are now inaccurate. So let's find out where data scientists live.

10. Create a bar graph of the respondants according to which country they are from.
Find the percentage of respondants from the top 5 countries

In [74]:

```
country = multipleChoiceResponses.groupby('Country').count()
sns.barplot(x=country.index, y='CurrentJobTitleSelect', data=country).set_title('respondant
plt.xticks(rotation='vertical')
plt.xlabel('Country')
plt.ylabel('Number')
plt.show()
```



In [75]:

```
country = multipleChoiceResponses.groupby(['Country'])
country = country[['CurrentJobTitleSelect']].count()
Percentage=country.apply(lambda x:100* x/x.sum()).reset_index()
round(Percentage.sort_values('CurrentJobTitleSelect',ascending=False)[:5],2)
```

Out[75]:

	Country	CurrentJobTitleSelect
47	United States	31.81
17	India	12.18
46	United Kingdom	4.27
13	Germany	3.67
4	Brazil	3.59

Answer

Percentage of respondants from top 5 countries:

1. United States 31.81%
2. India 12.18%
3. United Kingdom 4.27%
4. Germany 3.67%
5. Brazil 3.59%

11. What comments can you make about our previous comments on the United States, United Kingdom and Europe?

Are the majority of data scientists now likely to come from those countries?

Answer

Yes,the majority of data scientists come from United States,United Kingdom and Europe

12. Now that we have another demographic variable, let's see if there is any relationship between country, age and gender. We are specifically interested in the United States, India, United Kingdom, Germany and of course Australia!
Write code to output the mean and median age for United States, India, United Kingdom, Germany and Australia.

Hint: You may need to create a copy or slice.

In [76]:

```
# Your Code
country = multipleChoiceResponses[(multipleChoiceResponses['Country'] == 'United States') |
age = country.groupby(['Country', 'GenderSelect'])['Age'].agg(['mean', 'median']).reset_inc
round(age,2)
```

Out[76]:

	Country	GenderSelect	mean	median
0	Australia	Female	35.00	34
1	Australia	Male	37.16	36
2	Germany	Female	31.43	29
3	Germany	Male	36.63	34
4	India	A different identity	22.00	22
5	India	Female	29.06	28
6	India	Male	29.55	28
7	United Kingdom	A different identity	36.00	36
8	United Kingdom	Female	33.64	33
9	United Kingdom	Male	35.81	33
10	United States	A different identity	38.73	43
11	United States	Female	34.37	31
12	United States	Male	36.91	34

13. What Pattern do you notice about the relationship between age, gender for each of these countries?

Answer

In each of these countries, females are younger than their male counterparts. It may suggest that females are relatively new in this profession than males.

2. Education

So far we have seen that there may be some relationships between age, gender and the country that the respondents are from. Next, we should look at what their education is like.

2.1 Formal education

We saw in a recent activity that a significant number of job advertisements call for a masters degree or a PhD. Let's see if this is a reasonable ask based on the respondent's formal education.

14. Plot and display as text output the number and percentage of respondents with each type of formal

education.

In [77]:

```
# Your code
formalEducation = multipleChoiceResponses.groupby(['FormalEducation'])
formalEducation = formalEducation[['CurrentJobTitleSelect']].count().reset_index()
formalEducation.rename(columns={'CurrentJobTitleSelect':'number'},inplace=True)
print(formalEducation)
```

```
                FormalEducation  number
0             Bachelor's degree     930
1               Doctoral degree     808
2  Incomplete university studies      87
3               Master's degree    1594
4          No Formal education      25
5          Professional degree      96
```

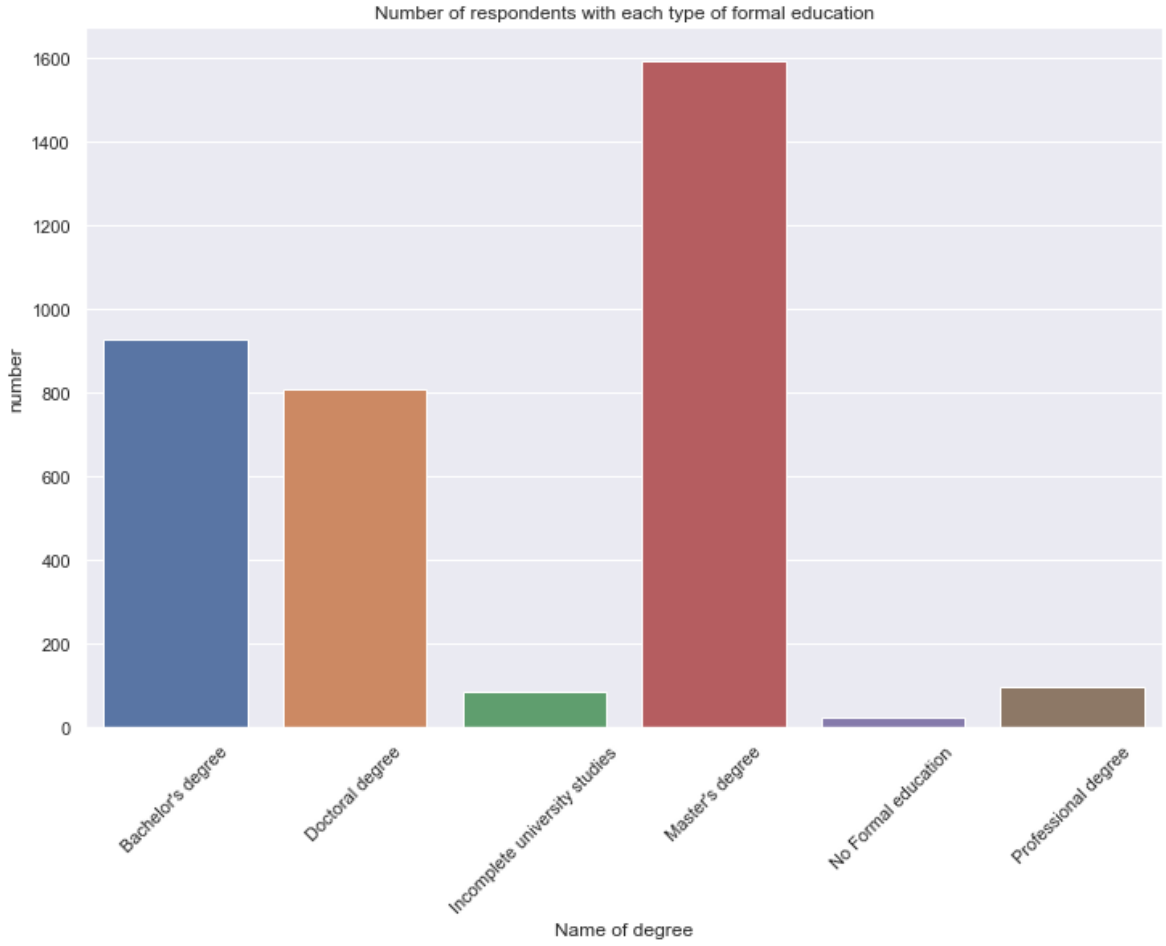
In [78]:

```
formalEducation = multipleChoiceResponses.groupby(['FormalEducation'])
formalEducation = formalEducation[['CurrentJobTitleSelect']].count()
formalEducation.rename(columns={'CurrentJobTitleSelect':'percentage'},inplace=True)
percentage=formalEducation.apply(lambda x:100* x/x.sum()).reset_index()
print(round(percentage))
```

```
                FormalEducation  percentage
0             Bachelor's degree        26.0
1               Doctoral degree        23.0
2  Incomplete university studies         2.0
3               Master's degree        45.0
4          No Formal education         1.0
5          Professional degree         3.0
```

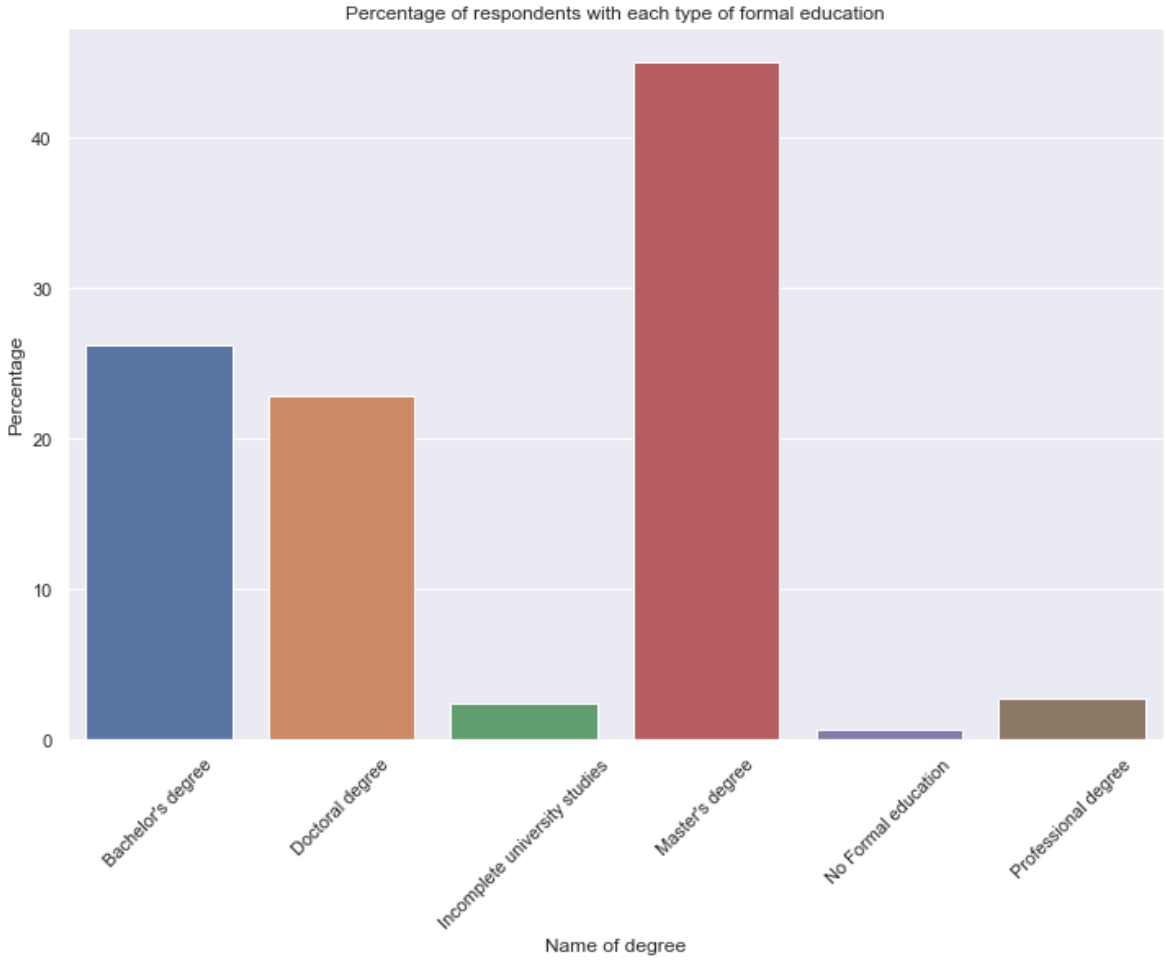
In [79]:

```
formalEducation = multipleChoiceResponses.groupby(['FormalEducation'])
formalEducation = formalEducation[['CurrentJobTitleSelect']].count().reset_index()

import numpy as np

ind = np.arange(len(formalEducation.FormalEducation))
fig,ax=plt.subplots()
sns.barplot(ind,formalEducation['CurrentJobTitleSelect'])

ax.set_xlabel('Name of degree')
ax.set_ylabel('number')
ax.set_title('Number of respondents with each type of formal education')
ax.set_xticks(ind)
ax.set_xticklabels(formalEducation['FormalEducation'],rotation=45)

fig.set_size_inches(12, 8)
```



In [80]:

```
formalEducation = multipleChoiceResponses.groupby(['FormalEducation'])
formalEducation = formalEducation[['CurrentJobTitleSelect']].count()
Percentage=formalEducation.apply(lambda x:100* x/x.sum()).reset_index()

import numpy as np

ind = np.arange(len(Percentage.FormalEducation))
fig,ax=plt.subplots()
sns.barplot(ind,Percentage['CurrentJobTitleSelect'])

ax.set_xlabel('Name of degree')
ax.set_ylabel('Percentage')
ax.set_title('Percentage of respondents with each type of formal education')
ax.set_xticks(ind)
ax.set_xticklabels(Percentage['FormalEducation'],rotation=45)

fig.set_size_inches(12, 8)
```



15. Based on what you have seen, do you think that a Master's or Doctoral degree is too unrealistic for job advertisers looking for someone with data science skills?

Give your reasons.

Answer

According to my observation,I've come to a conclusion that a Master's degree plays an important role for data science skills.But Doctoral degree is not too much unrealistic for job advertisers looking for someone with data science skills.

16. Let's see if the trend is reflected in the Australian respondents.
Plot and display as text output the number and percentage of Australian respondents with each type of formal education.

In [81]:

```
# Your code
country=multipleChoiceResponses.set_index('Country')
australia=country.loc['Australia']
formalEducation = australia.groupby(['FormalEducation'])
formalEducation = formalEducation[['CurrentJobTitleSelect']].count().reset_index()
formalEducation.rename(columns={'CurrentJobTitleSelect':'number'},inplace=True)
print(formalEducation)
```

```
                FormalEducation  number
0             Bachelor's degree      45
1               Doctoral degree      25
2  Incomplete university studies       5
3               Master's degree      42
4           Professional degree       2
```

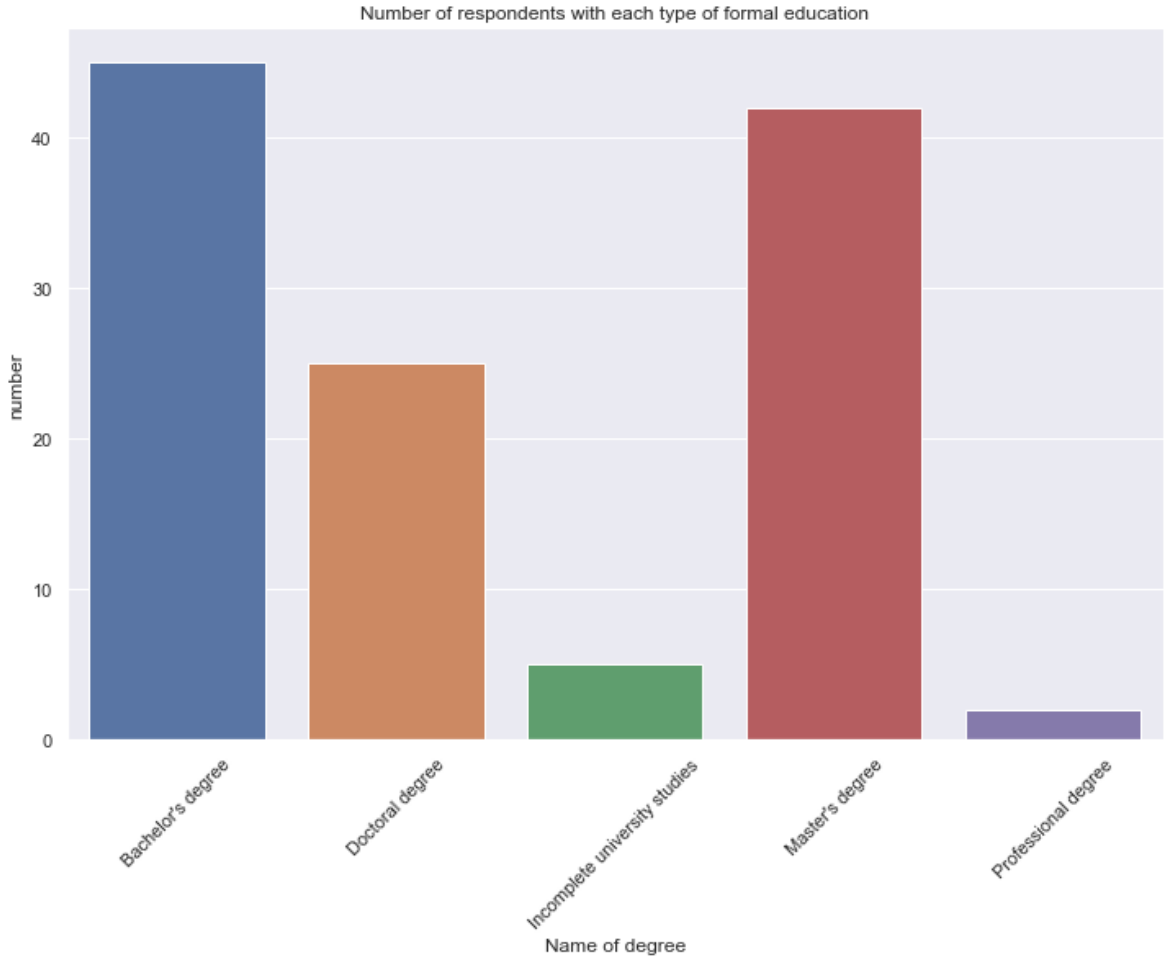
In [82]:

```
country=multipleChoiceResponses.set_index('Country')
australia=country.loc['Australia']
formalEducation = australia.groupby(['FormalEducation'])
formalEducation = formalEducation[['CurrentJobTitleSelect']].count()
formalEducation.rename(columns={'CurrentJobTitleSelect':'percentage'},inplace=True)
percentage=formalEducation.apply(lambda x:100* x/x.sum()).reset_index()
print(round(percentage))
```

```
                FormalEducation  percentage
0             Bachelor's degree        38.0
1               Doctoral degree        21.0
2  Incomplete university studies         4.0
3               Master's degree        35.0
4           Professional degree         2.0
```

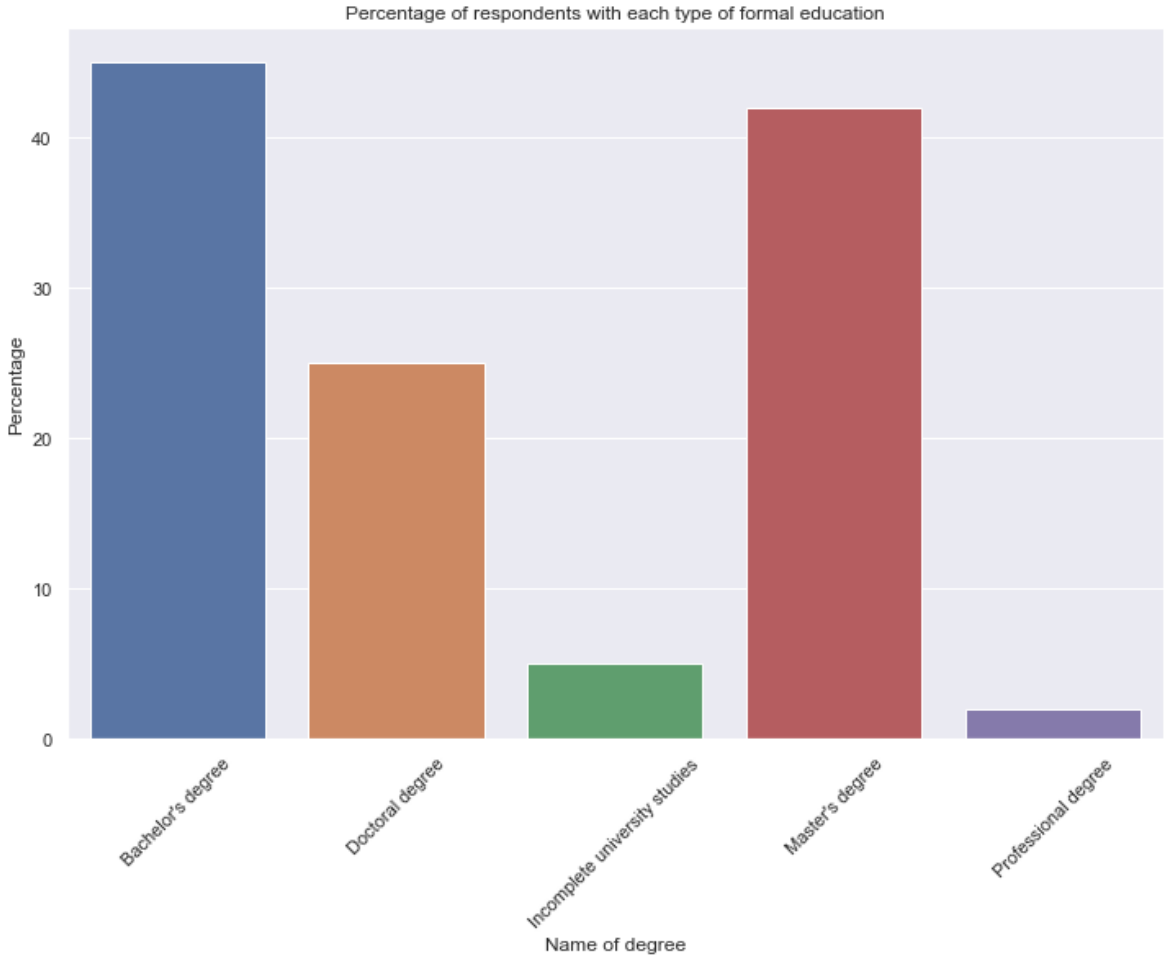
In [83]:

```
country=multipleChoiceResponses.set_index('Country')
australia=country.loc['Australia']
formalEducation = australia.groupby(['FormalEducation'])
formalEducation = formalEducation[['CurrentJobTitleSelect']].count().reset_index()

import numpy as np

ind = np.arange(len(formalEducation.FormalEducation))
fig,ax=plt.subplots()
sns.barplot(ind,formalEducation['CurrentJobTitleSelect'])
ax.set_xlabel('Name of degree')
ax.set_ylabel('number')
ax.set_title('Number of respondents with each type of formal education')
ax.set_xticks(ind)
ax.set_xticklabels(formalEducation['FormalEducation'],rotation=45)

fig.set_size_inches(12, 8)
```



In [84]:

```
country=multipleChoiceResponses.set_index('Country')
australia=country.loc['Australia']
formalEducation = australia.groupby(['FormalEducation'])
formalEducation = formalEducation[['CurrentJobTitleSelect']].count()
Percentage=formalEducation.apply(lambda x:100* x/x.sum()).reset_index()

import numpy as np

ind = np.arange(len(Percentage.FormalEducation))
fig,ax=plt.subplots()
sns.barplot(ind,formalEducation['CurrentJobTitleSelect'])

ax.set_xlabel('Name of degree')
ax.set_ylabel('Percentage')
ax.set_title('Percentage of respondents with each type of formal education')
ax.set_xticks(ind)
ax.set_xticklabels(Percentage['FormalEducation'],rotation=45)

fig.set_size_inches(12, 8)
```



17. Display as text output the mean and median age of each respondant according to each degree type.

In [85]:

```
# Your code
age = multipleChoiceResponses.groupby('FormalEducation')['Age'].agg(['mean', 'median'])
age
```

Out[85]:

FormalEducation	mean	median
Bachelor's degree	30.632258	28.0
Doctoral degree	39.235149	37.0
Incomplete university studies	36.011494	35.0
Master's degree	33.746550	31.0
No Formal education	41.680000	42.0
Professional degree	36.645833	34.5

3. Employment

After you complete your degree many of you will be seeking work. The graduate employment four months after graduation in Australia is 69.5%. At Monash, it is 70.1%. This is for all Australian degrees. Let's have a look at the state of the employment market for the respondents.

Let's have a look at the data.

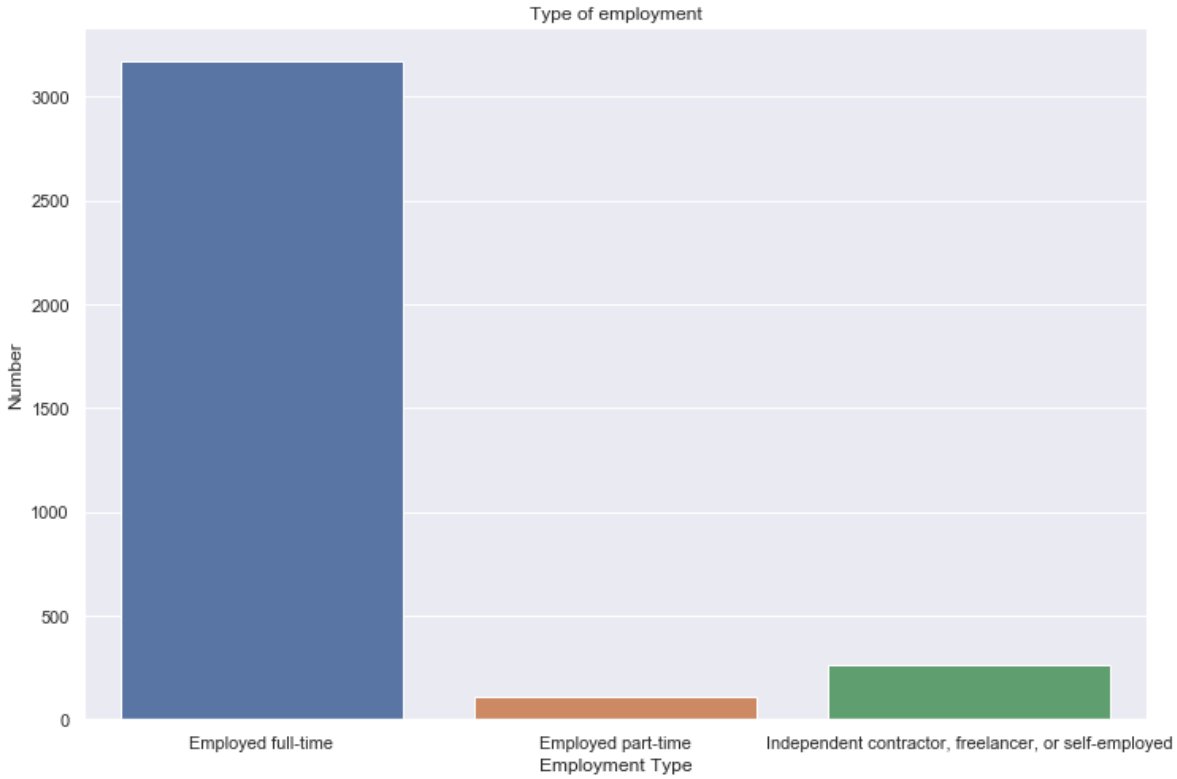
3.1 Employment status

The type of employment will affect the salary of a worker. Those employed part-time will likely earn less than those who work full time.

18. Plot the type of employment the respondents have on a bar chart.

In [86]:

```
# Your code
employment = multipleChoiceResponses.groupby('EmploymentStatus').count()
sns.barplot(x=employment.index, y='Age', data=employment).set_title('Type of employment')
plt.xlabel('Employment Type')
plt.ylabel('Number')
plt.show()
```

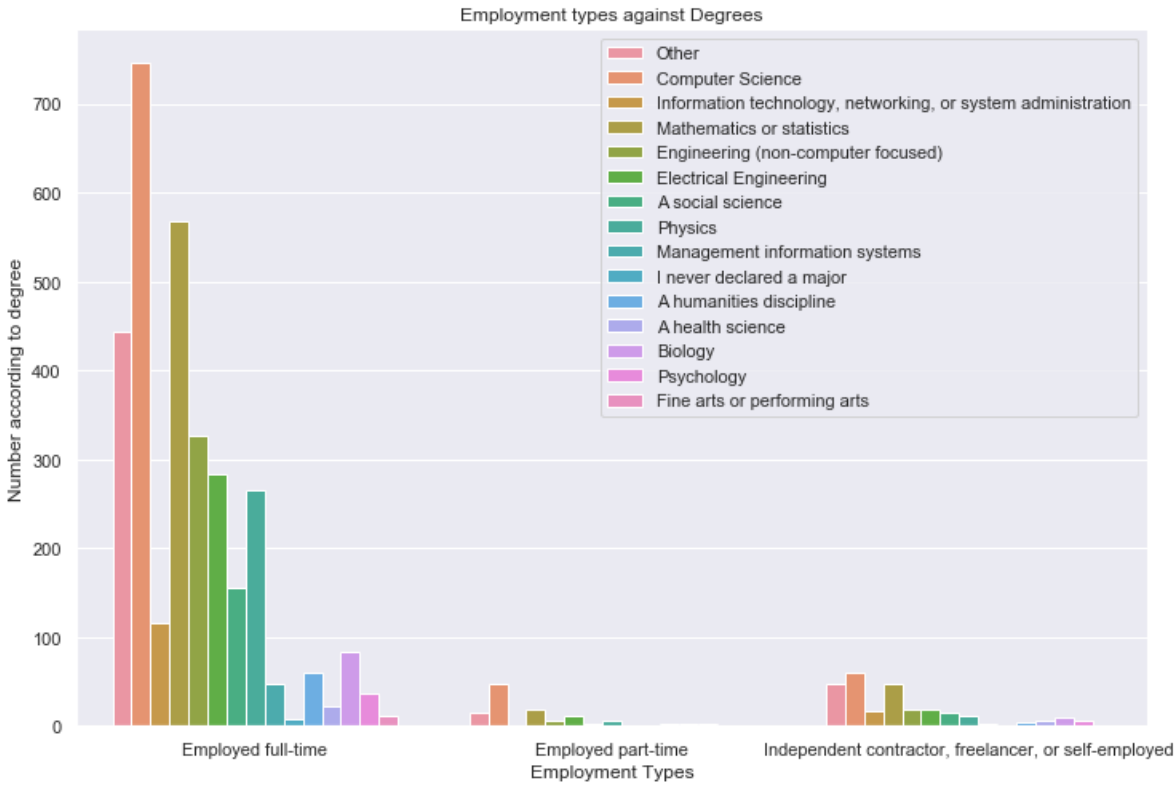


19. You may be wondering if your own degree and experince will help you gain full time employment after you graduate.

Plot the respondants employment types against their degrees.

In [87]:

```
# Your code
sns.countplot(x='EmploymentStatus', hue='MajorSelect', data=multipleChoiceResponses).set_ti
plt.ylabel('Number according to degree')
plt.xlabel('Employment Types')
plt.legend(loc='upper right')
plt.show()
```



20. Looking at the graph, which degree is best to gain full-time employment?
What is odd about IT, networking or system administration??

Explain your answers.

Answer From the graph, it is evident that a degree in computer science gives anyone the highest possibility of landing a full-time employment.

Degree holders in IT, networking, or system administration seems to have the lowest number of jobs, be that full-time, part-time, or freelance.

21. Overall, we know that 92.71% of respondents are employed, and 89.55% are employed full time. This may not be the same for every country. Print out the percentages of all respondents who are employed full time in Australia, United Kingdom and the United States.

In [88]:

```
# Your code
employment = multipleChoiceResponses.set_index('EmploymentStatus')
country = employment.loc['Employed full-time']

country.set_index('Country',inplace= True)
australia = country.loc['Australia']
uk=country.loc['United Kingdom']
usa=country.loc['United States']

emp = australia.groupby(['FormalEducation'])
emp = emp[['CurrentJobTitleSelect']].count()
emp.rename(columns={'CurrentJobTitleSelect':'percentage'},inplace=True)
AUS_Percentage=emp.apply(lambda x:100* x/x.sum()).reset_index()
AUS_Percentage
```

Out[88]:

	FormalEducation	percentage
0	Bachelor's degree	39.603960
1	Doctoral degree	19.801980
2	Incomplete university studies	3.960396
3	Master's degree	34.653465
4	Professional degree	1.980198

In [89]:

```
emp = uk.groupby(['FormalEducation'])
emp = emp[['CurrentJobTitleSelect']].count()
emp.rename(columns={'CurrentJobTitleSelect':'percentage'},inplace=True)
UK_Percentage=emp.apply(lambda x:100* x/x.sum()).reset_index()
UK_Percentage
```

Out[89]:

	FormalEducation	percentage
0	Bachelor's degree	21.897810
1	Doctoral degree	36.496350
2	Incomplete university studies	0.729927
3	Master's degree	40.145985
4	Professional degree	0.729927

In [90]:

```
emp = usa.groupby(['FormalEducation'])
emp = emp[['CurrentJobTitleSelect']].count()
emp.rename(columns={'CurrentJobTitleSelect':'percentage'},inplace=True)
USA_Percentage=emp.apply(lambda x:100* x/x.sum()).reset_index()
USA_Percentage
```

Out[90]:

	FormalEducation	percentage
0	Bachelor's degree	24.174757
1	Doctoral degree	27.572816
2	Incomplete university studies	1.941748
3	Master's degree	44.368932
4	No Formal education	0.485437
5	Professional degree	1.456311

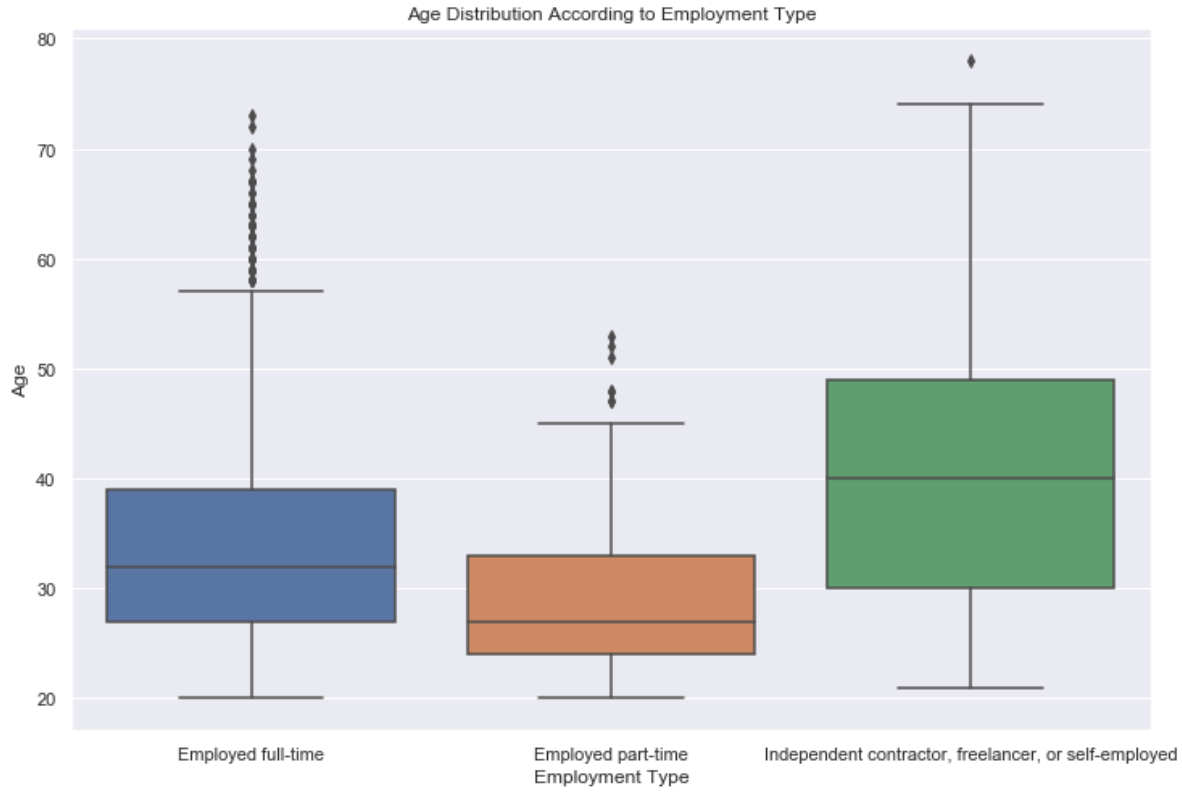
Remember earlier we saw that age seemed to have some interesting characteristics when plotted with other variables.

Let's find out the median age of employees by type of employment.

22. Plot a boxplot of the respondents age grouped by employment type.

In [91]:

```
# Your code
age_median =  multipleChoiceResponses.groupby('EmploymentStatus')['Age']
sns.boxplot(x="EmploymentStatus", y="Age", data= multipleChoiceResponses).set_title('Age Di
plt.xlabel('Employment Type')
plt.show()
```



Now this is interesting, full time employees seem to be a little older than part time employees. Independent contactors, freelancers and self-employed respondents are older still.

4. Salary

Data science is considered a very well paying role and was named 'best job of the year' for 2016.

We had a look around and saw that data scientists were paid between $110,823 at IBM and 149,963 at Apple, in Australian dollars.

On average it seems that $116,840 is what an Australian Data scientist can expect to earn. Do you think this is reasonable? Is this any different to the rest of the world?

4.1 Salary overview

Since all of the respondents did not come from one country, we can assume that they gave their salaries in their countries currency. We have filtered the data for you and provided exchange rates in a file called *conversionRates.csv* which should already be imported.

Let's have a look at the data.

23. Use the codes for each country to merge the files so that you can convert the salary data to Australian Dollars (AUD). Print out the maximum and median salary in AUD. Hint: think about what data type you have.

In [92]:

```
# Your code
conversion=pd.melt(conversionRates,id_vars=['originCountry'],value_vars=['exchangeRateAUS']
conversion.rename(columns = {'originCountry':'CompensationCurrency'}, inplace = True)

Salary=pd.merge(conversion,multipleChoiceResponses,on=['CompensationCurrency'])
Salary['SalaryAUD']=Salary['CompensationAmount']*Salary['value']

salary = np.array(Salary['SalaryAUD'])
print("Median Salary :", round(Salary.SalaryAUD.median()))
print("Maximum Salary :", round(salary.max()))
```

```
Median Salary : 76998
Maximum Salary : 790290.0
```

24. Do those figures reflect the values at the beginning of this section? Why do you think so?

Answer

I think those values don't reflect at the beginning of this section.Because it is mentioned that data scientists were paid between AU$110,823 at IBM and AU$149,963 at Apple.But we can see that the median salary is AU$76998 which is much different from given data.

4.2 Salary by country

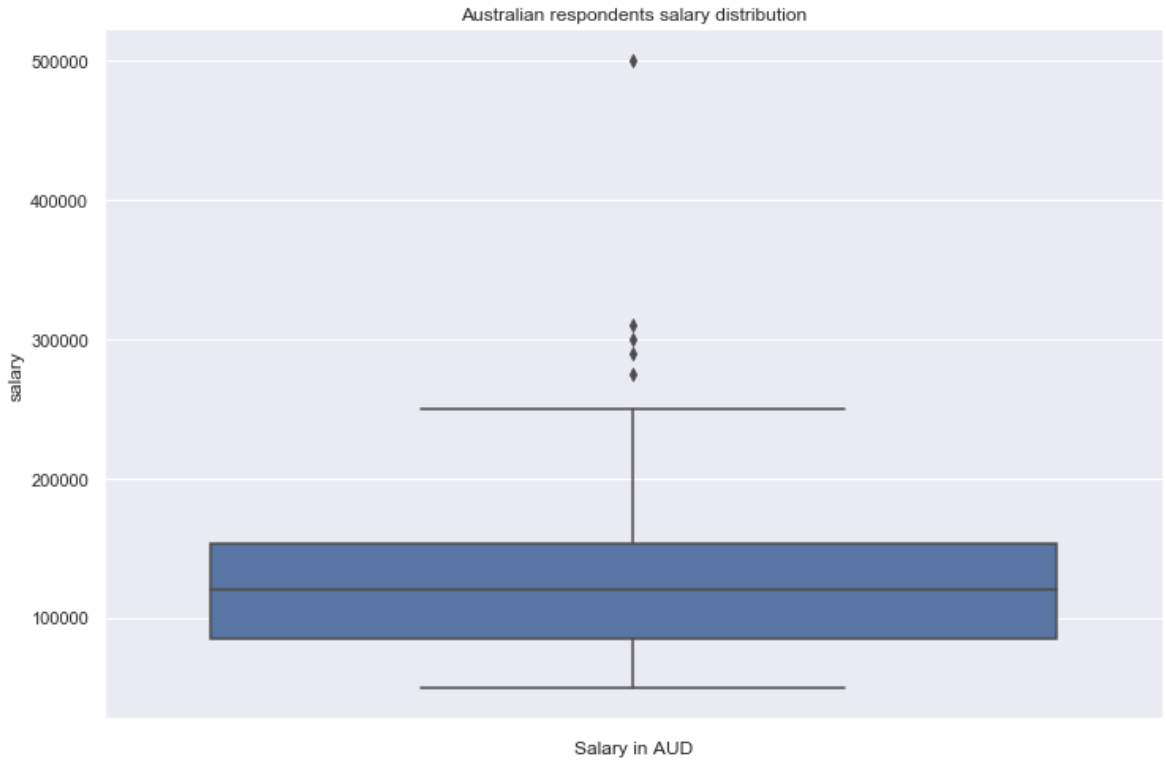
Since each country has different cost of living and pay indexes, we should see how they compare.

25. Plot a boxplot of the Australian respondents salary distribution. Print out the maximum and median salaries for Australian repsondnts.

In [93]:

```
# Your code
AusSalary=Salary.set_index('Country')
AUD=AusSalary.loc['Australia']
print("Maximum Salary for Australian Respondnts:", AUD.loc[:,'SalaryAUD'].max())
print("Median Salary for Australian Respondnts:", AUD.loc[:,'SalaryAUD'].median())

sns.boxplot(y='SalaryAUD',data=AUD).set_title("Australian respondents salary distribution")
plt.xlabel('Salary in AUD')
plt.ylabel('salary')
plt.show()
```

```
Maximum Salary for Australian Respondnts: 500000.0
Median Salary for Australian Respondnts: 120000.0
```



26. Do those figures for Australia reflect the values at the beginning of this section?

Answer

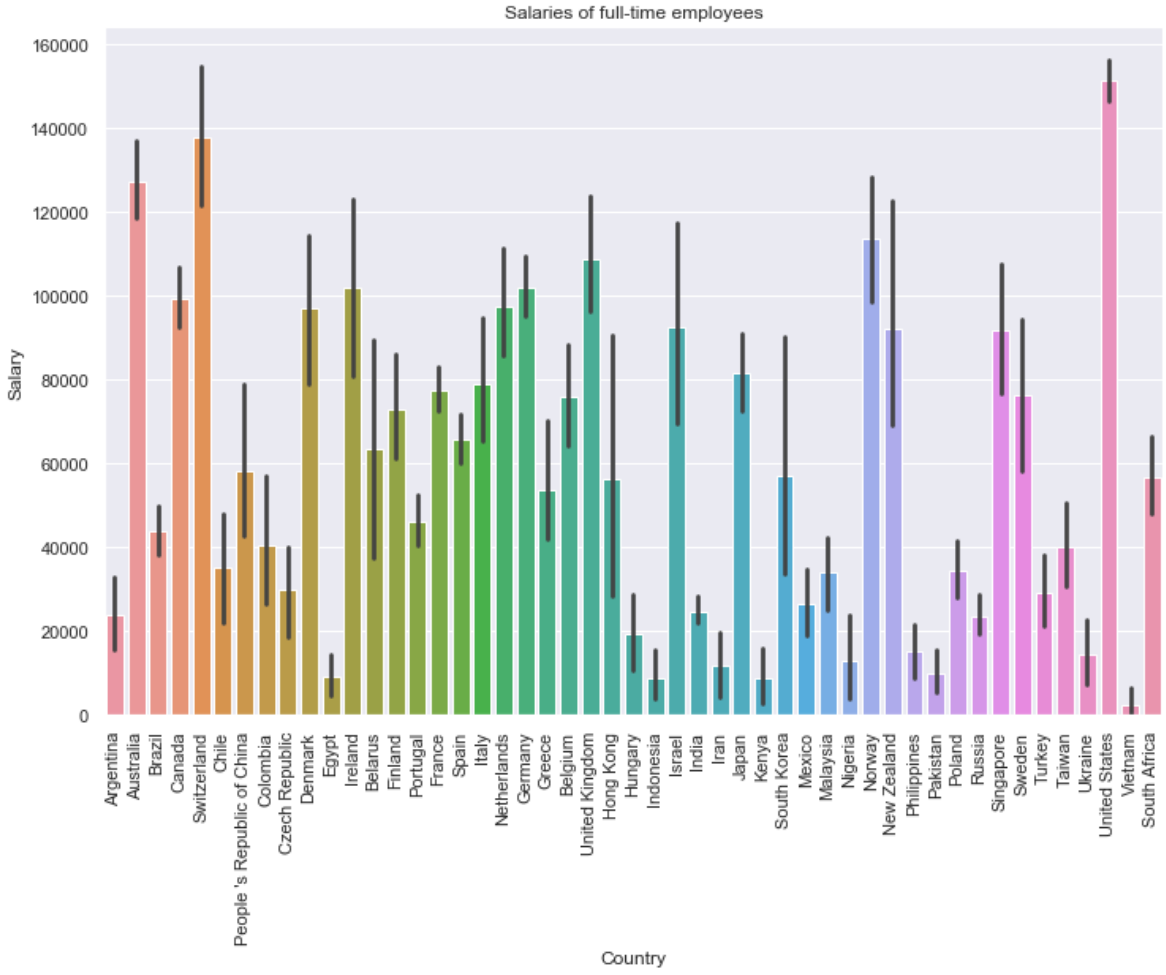
Yes,those figures for Australia reflect the values at the beginning of this section.

27. Australia's salaries look pretty good.
Plot the salaries of all countries on a bar chart.

Hint: Adjust for full-time employees only

In [94]:

```
# Your code
salary=Salary.set_index('EmploymentStatus')
allSalary=salary.loc['Employed full-time']
sns.barplot(x='Country',y='SalaryAUD',data=allSalary).set_title('Salaries of full-time empl

plt.xlabel("Country")
plt.ylabel("Salary")
plt.xticks(rotation="vertical")
plt.show()
```



28. What do you notice about the distributions? What do you think is the cause of this?

Answer Country with better pay index and high living cost tends to spend more for data scientists. For example, United States, United Kingdom, Australia, Switzerland, etc, are among the top paying countries. On the contrary, though a large number of respondants were from India, they are paid much less due to their lower pay index and living cost.

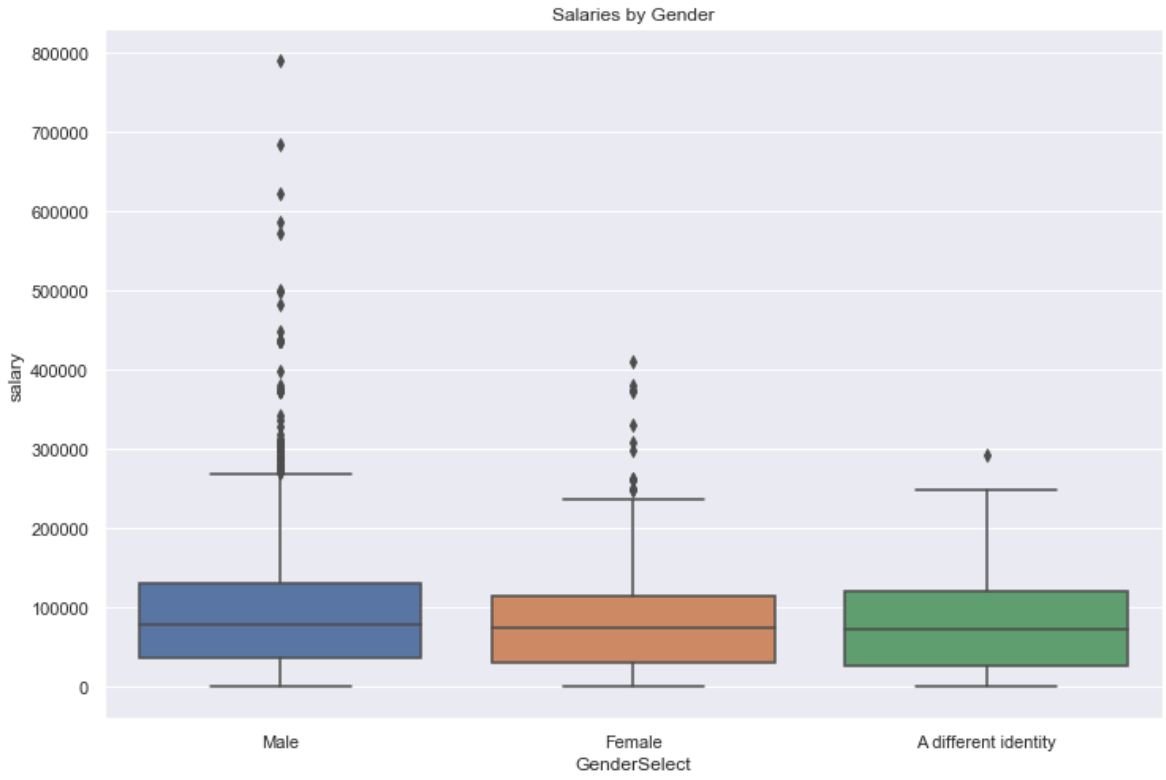
4.3 Salary and Gender

The gender pay gap in the tech industry is a big talking point. Let's see if the respondents are noticing the effect.

29. Plot the salaries of all countries grouped by gender on a boxplot.

In [95]:

```
# Your code
sns.boxplot(y='SalaryAUD',x='GenderSelect',data=Salary).set_title('Salaries by Gender')
plt.ylabel('salary')
plt.show()
```



30. What do you notice about the distributions?

Answer

Males have the highest amount of salary compared to female and different identity genders.The mean and median salary of three gender identity are nearly same.

31. The salaries may be affected by the country the responant is from. In Australia the weekly difference in pay between men and women is 17.7% and in the United states it is 26%.
Print the median salaries of Australia, United States and India grouped by gender.

In [96]:

```
# Your code
medianSalary = Salary[(Salary['Country'] == 'Australia') | (Salary['Country'] == 'United St
medianSalary.groupby(['Country', 'GenderSelect'])['SalaryAUD'].median()
```

Out[96]:

```
Country        GenderSelect
Australia      Female                  82000.000000
               Male                   130000.000000
India          A different identity    13628.148800
               Female                  12654.709600
               Male                    17327.217760
United States  A different identity   168264.137295
               Female                 112176.091530
               Male                   143336.116955
Name: SalaryAUD, dtype: float64
```

4.4 Salary and formal education

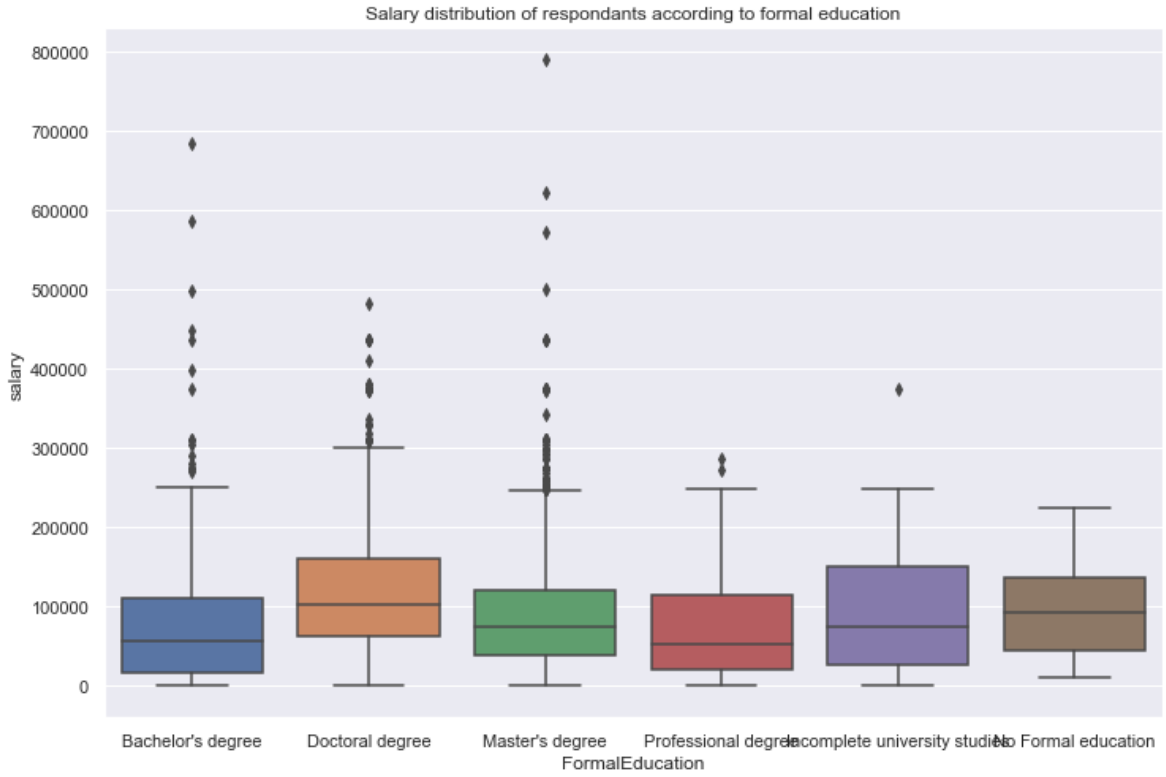
Is getting your master's really worth it ? Do PhDs get more money?

Let's see.

32. Plot the salary distribution of all respondants and group by formal education type on a boxplot.

In [97]:

```
# Your code
plt.figure(figsize=(12, 8))
sns.boxplot(y='SalaryAUD',x='FormalEducation',data=Salary).set_title('Salary distribution c
plt.ylabel('salary')
plt.show()
```



33. Is it better to get your Masters or PhD?
Explain your answer.

Answer Yes,it is better to get Masters or PhD.Because both Masters and PhD holders have the mean salary which is greater than compared to others.And a Master holder is getting highest salary among the respondnts.

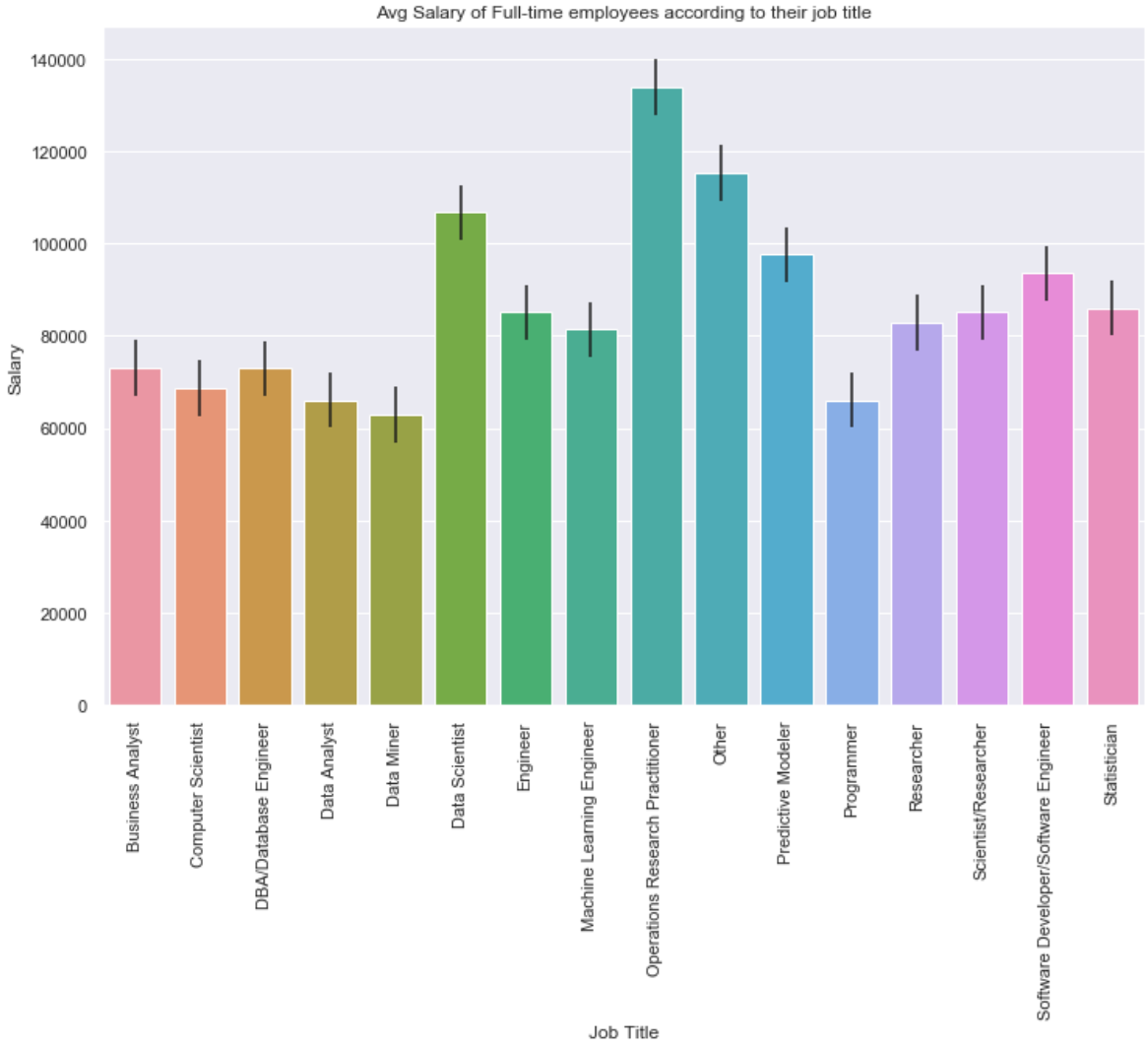
4.5 Salary and job

So are data scientists the highest paid in the industry? Or are there lesser known roles that are hiding from the spotlight?

34. Plot a bar chart of average salary (with error bars) of full time employees and group by job title.

In [98]:

```
# Your code
avgSalary=Salary.set_index('EmploymentStatus')
Avg=avgSalary.loc['Employed full-time']
sal=Avg.groupby(['CurrentJobTitleSelect'])['SalaryAUD'].mean()
sns.barplot(x=sal.index,y=sal,data=Salary,yerr=6000).set_title("Avg Salary of Full-time emp
plt.xlabel("Job Title")
plt.ylabel("Salary")
plt.xticks(rotation="vertical")
plt.show()
```



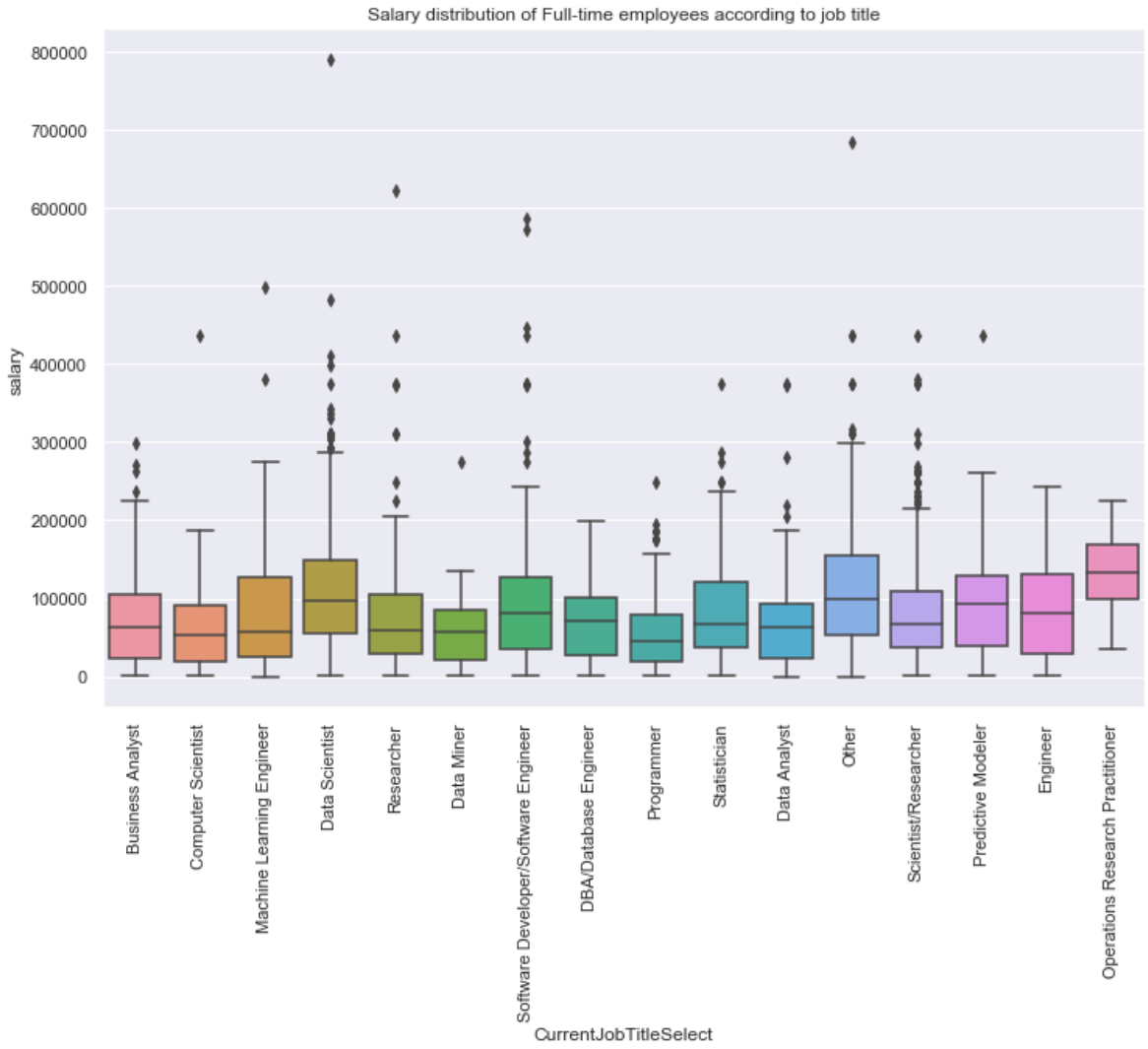
35. Which job earns the most? Give a brief explaination of that job.

Answer Operations Reseach Practitioner earns the most.

36. So why are data scientists in the spotlight? Plot the salary distribution of full-time employees and group by job title as boxplots.

In [99]:

```
# Your code
salary=Salary.set_index('EmploymentStatus')
sal=salary.loc['Employed full-time']
sns.boxplot(y='SalaryAUD',x='CurrentJobTitleSelect',data=sal).set_title('Salary distributic
plt.xticks(rotation='vertical')
plt.ylabel('salary')
plt.show()
```



37. Do the boxplots give some insight into why data scientists may receive so much attention? Explain your answer.

Answer Yes,the boxplots give some insight into why data scientists may receive so much attention. A data scientist is getting highest salary among all the respondnts.

5. Predicting salary

We have looked at many variables and seen that there are a lot of factors that could affect your salary.

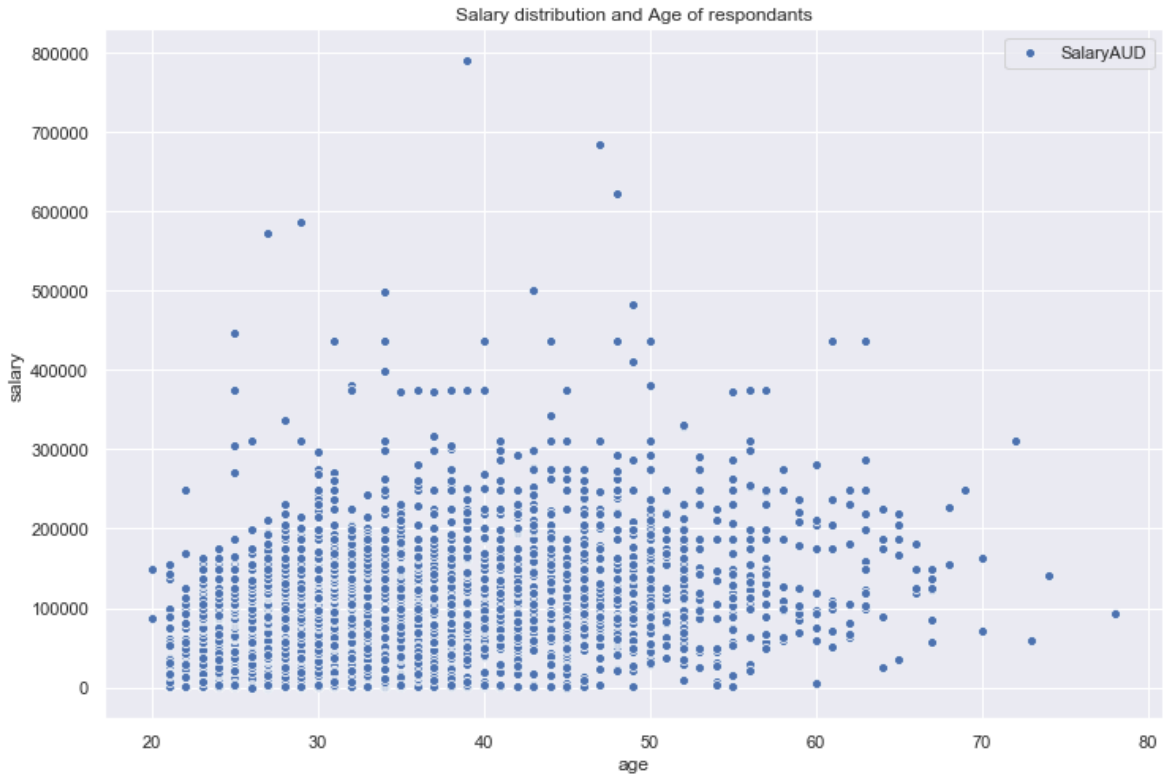
Let's say we wanted to reduce it though? One method we could use is a linear regression. This is a very basic model that can give us some insights. Note though, there are more robust ways to predict salary based on categorical variables. But this exercise will give you a taste of predictive modelling.

38. Plot the salary distribution and age of respondants on a scatterplot.

In [100]:

```
# Your code
sns.scatterplot('Age','SalaryAUD',data=Salary,label="SalaryAUD").set_title('Salary distribu
ax.legend(loc='best')
plt.xlabel("age")
plt.ylabel("salary")
plt.show()
```

```
No handles with labels found to put in legend.
```



39. There may be a weak relationship. Let's refine this.

Create a linear regression between the salary and age of full-time Australian respondants. Plot the linear fit over the scatterplot.

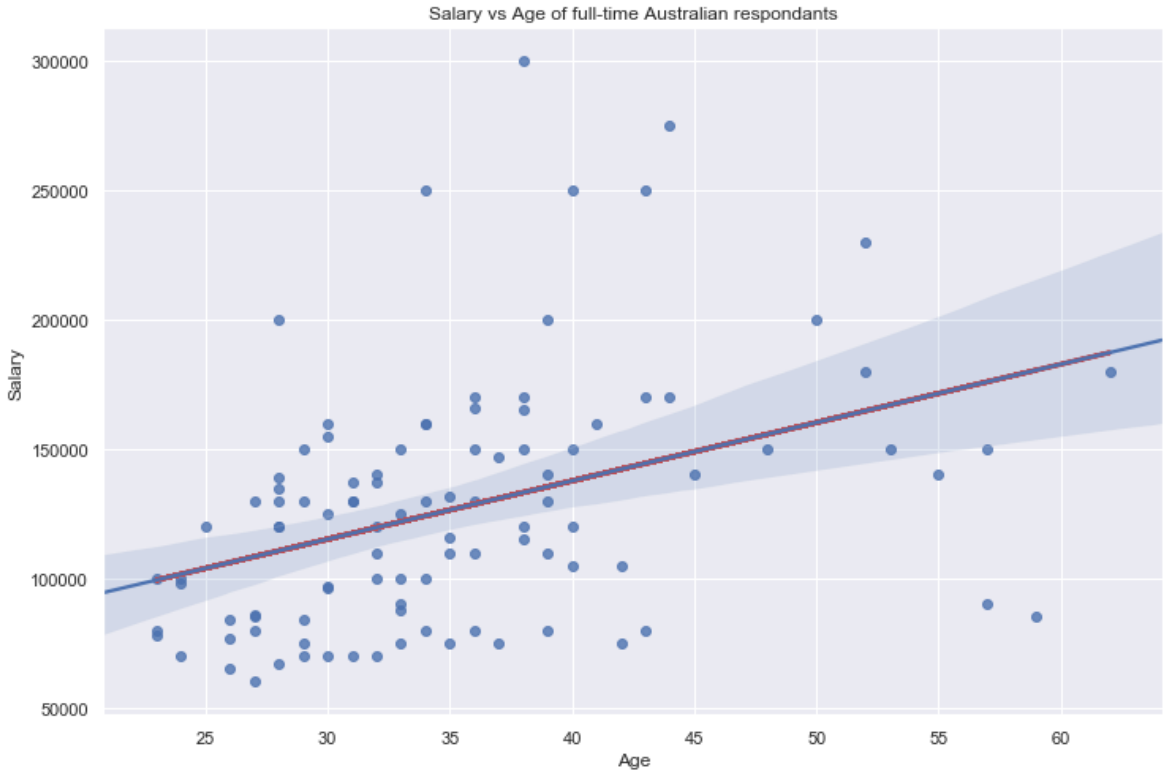
In [101]:

```
#Your code
employment = Salary.set_index('EmploymentStatus')
country = employment.loc['Employed full-time']

country.set_index('Country',inplace= True)
australia = country.loc['Australia']


from scipy.stats import linregress

slope, intercept, r_value, p_value, std_err = linregress( australia['Age'],  australia['Sal

line = [slope*xi + intercept for xi in  australia['Age']]

plt.plot( australia['Age'], line,'r-', linewidth=3)

sns.regplot(australia['Age'], australia['SalaryAUD'],label="SalaryAUD").set_title('Salary v
ax.legend(loc='best')
plt.xlabel('Age')
plt.ylabel('Salary')
```

```
No handles with labels found to put in legend.
```

Out[101]:

```
Text(0, 0.5, 'Salary')
```



40. Do You think that this is a good way to predict salaries?
Explain your answer.

Answer

Yes.Because from this regression line I can predict that salary range of most of the full-time Australian respondants according to their age.

Well done you have completed Part A. Don't forget Part B below.

For reassurance, the Graduate Careers Australia 2016 survey found the median salary for masters graduates in Computer Science and IT was $76,000.

Task B - Exploratory Analysis on Other Data

Find some publicly available data and repeat some of the analysis performed in Task A above. Good sources of data are government websites, such as: data.gov.au, data.gov, data.gov.in, data.gov.uk, ...

Please note that your report and analysis should contain consideration of the data you have found and its broader impact in terms of (1) the purpose of the data, (2) ethics and privacy issues, (3) environmental impact, (4) societal benefit, (5) health benefit, and (6) commercial benefit. Moreover, your analysis should at least involve (7) visualisation, (8) interpretation of your visualisation and (9) a prediction task.

To perform Task B, you can continue by extending this jupyter notebook file by adding more cells.

Students Performance in Exams. This dataset is taken from kaggle.com which consists of the marks obtained by students in their exams.The file is 'StudentsPerformance.csv'.

Import the file

In [102]:

```
studentperformance=pd.read_csv('StudentsPerformance.csv')
studentperformance
```

Out[102]:

	gender	ethnicity	parentalEducation	lunch	TestPreparation	MathScore	ReadingScore	WritingScc
0	female	group B	bachelor's degree	standard	none	72	72	
1	female	group C	some college	standard	completed	69	90	
2	female	group B	master's degree	standard	none	90	95	
3	male	group A	associate's degree	free/reduced	none	47	57	
4	male	group C	some college	standard	none	76	78	
5	female	group B	associate's degree	standard	none	71	83	
6	female	group B	some college	standard	completed	88	95	
7	male	group B	some college	free/reduced	none	40	43	
8	male	group D	high school	free/reduced	completed	64	64	

1. Find number and percentage of each gender.

In [103]:

```
Gender = studentperformance.groupby(['gender'])
gender = Gender[['lunch']].count().reset_index()
gender.rename(columns={'lunch':'number'},inplace=True)
gender
```

Out[103]:

	gender	number
0	female	518
1	male	482

In [104]:

```
Gender = studentperformance.groupby(['gender'])
gender = Gender[['lunch']].count()
gender.rename(columns={'lunch':'percentage'},inplace=True)
Percentage=gender.apply(lambda x:100* x/x.sum()).reset_index()
round(Percentage,2)
```

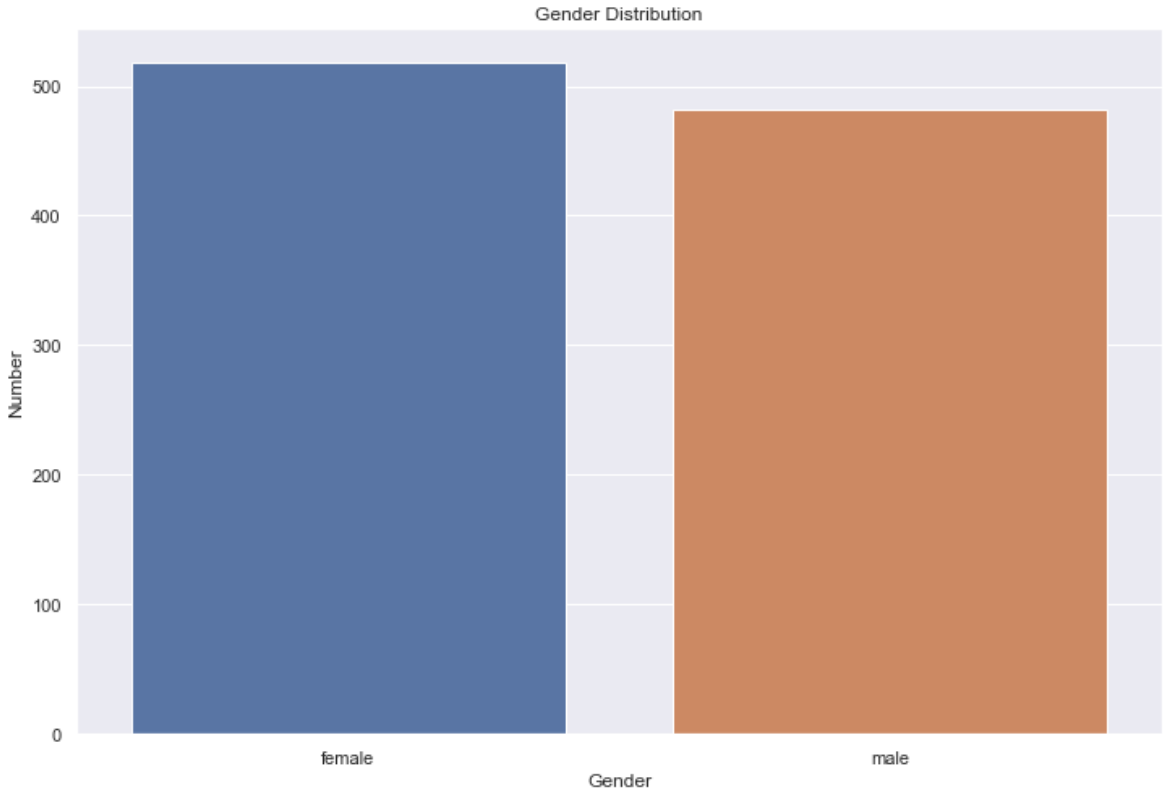
Out[104]:

	gender	percentage
0	female	51.8
1	male	48.2

2. Plot Gender distribution of the students

In [105]:

```
gender = studentperformance.groupby('gender').count()
sns.barplot(x=gender.index, y='lunch', data=gender).set_title('Gender Distribution')
plt.xlabel('Gender')
plt.ylabel('Number')
plt.show()
```



3.Find and plot number of the students according to their parental education.

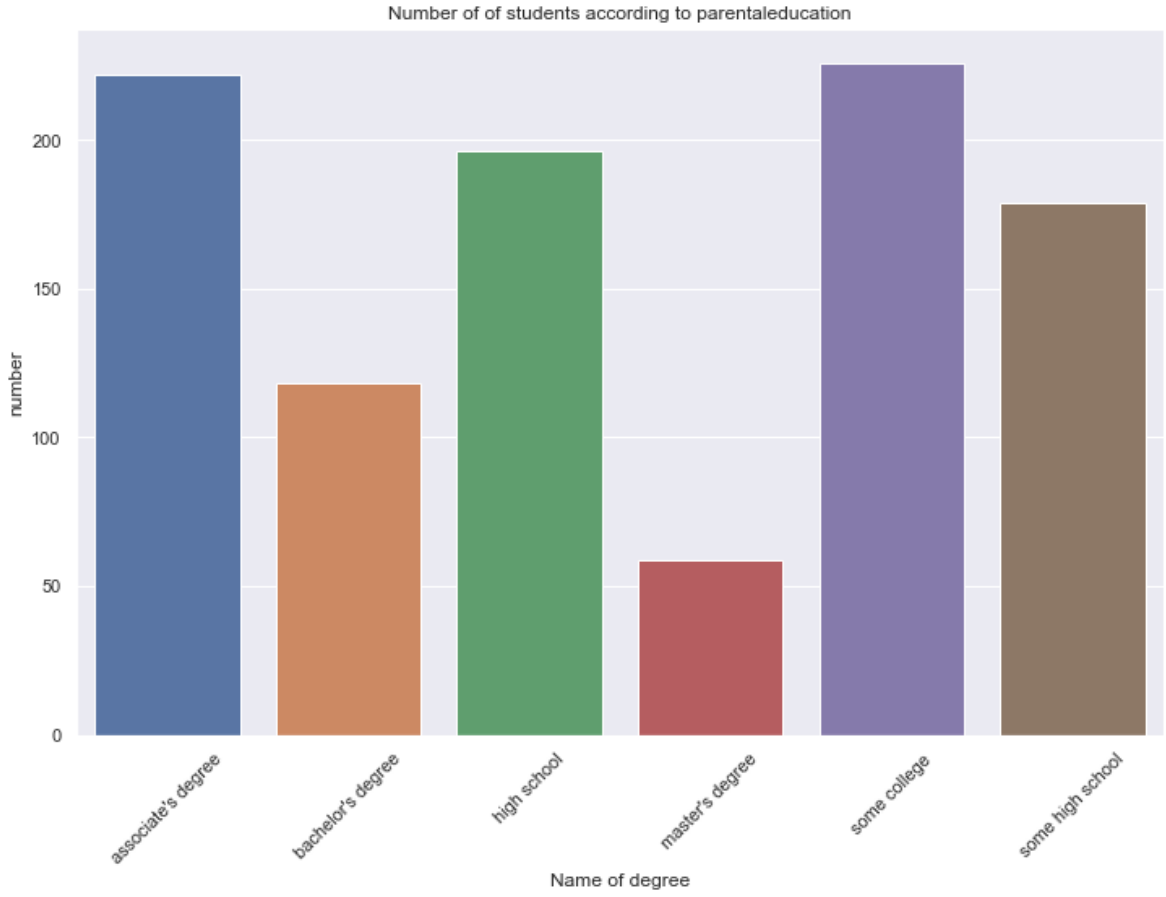
In [106]:

```
education = studentperformance.groupby(['parentalEducation'])
paternaleducation = education[['lunch']].count().reset_index()
paternaleducation.rename(columns={'lunch':'number'},inplace=True)
print(paternaleducation)
```

```
    parentalEducation  number
0  associate's degree     222
1   bachelor's degree     118
2         high school     196
3     master's degree      59
4        some college     226
5    some high school     179
```

In [107]:

```
education = studentperformance.groupby(['parentalEducation'])
paternaleducation = education[['lunch']].count().reset_index()

import numpy as np

ind = np.arange(len(paternaleducation.parentalEducation))
fig,ax=plt.subplots()
sns.barplot(ind,paternaleducation['lunch'])

ax.set_xlabel('Name of degree')
ax.set_ylabel('number')
ax.set_title('Number of of students according to parentaleducation')
ax.set_xticks(ind)
ax.set_xticklabels(paternaleducation['parentalEducation'],rotation=45)

fig.set_size_inches(12, 8)
```



4.Find and plot percentage of students according to their parental education.

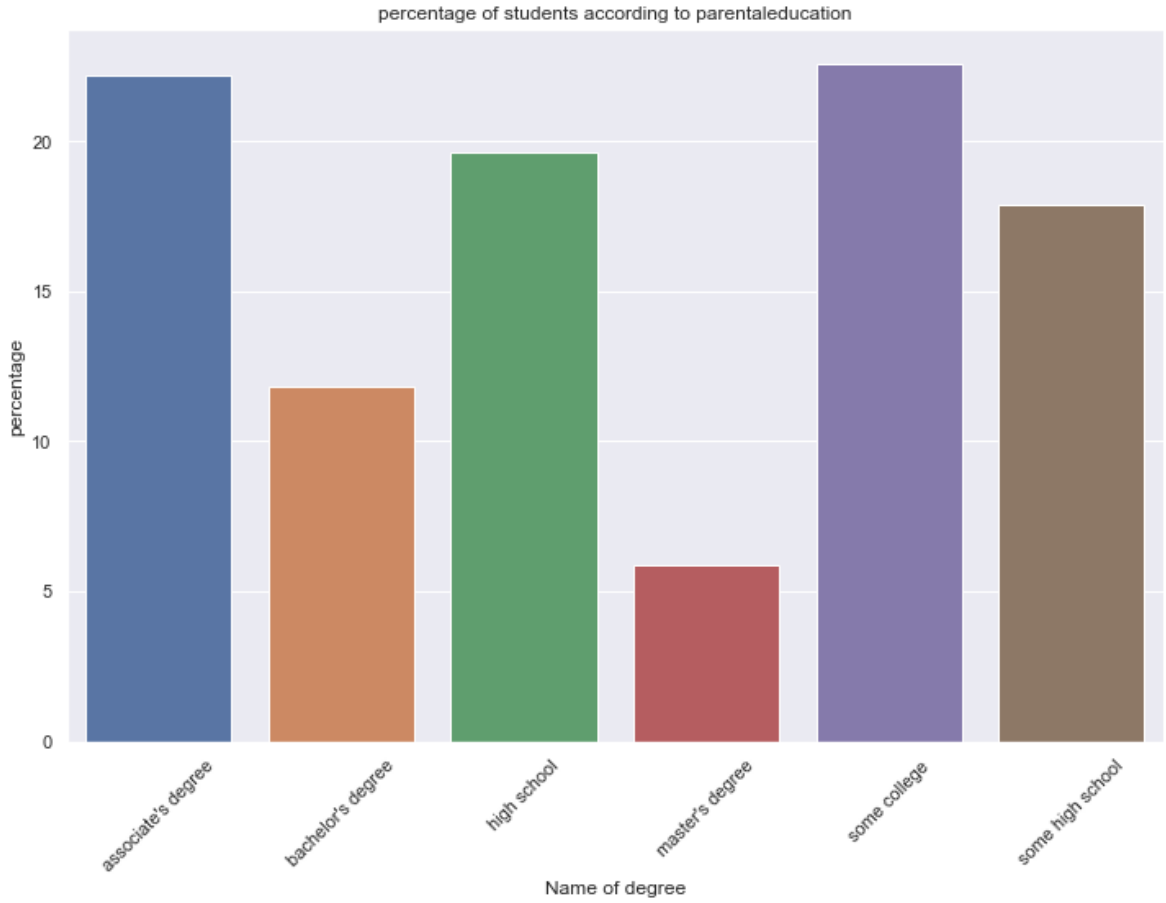
In [108]:

```
education = studentperformance.groupby(['parentalEducation'])
paternaleducation = education[['lunch']].count()
paternaleducation.rename(columns={'lunch':'number'},inplace=True)
percentage=paternaleducation.apply(lambda x:100* x/x.sum()).reset_index()
print(percentage)
```

```
    parentalEducation  number
0  associate's degree    22.2
1   bachelor's degree    11.8
2         high school    19.6
3     master's degree     5.9
4        some college    22.6
5    some high school    17.9
```

In [109]:

```
education = studentperformance.groupby(['parentalEducation'])
paternaleducation = education[['lunch']].count()
percentage=paternaleducation.apply(lambda x:100* x/x.sum()).reset_index()

import numpy as np

ind = np.arange(len(percentage.parentalEducation))
fig,ax=plt.subplots()
sns.barplot(ind,percentage['lunch'])

ax.set_xlabel('Name of degree')
ax.set_ylabel('percentage')
ax.set_title('percentage of students according to parentaleducation')
ax.set_xticks(ind)
ax.set_xticklabels(percentage['parentalEducation'],rotation=45)

fig.set_size_inches(12, 8)
```



5.Find number and percentage of students groupby their ethnicity. Also plot the number of students according to ethnicity.

In [110]:

```
race = studentperformance.groupby(['ethnicity'])
Ethnicity = race[['lunch']].count().reset_index()
Ethnicity.rename(columns={'lunch':'number'},inplace=True)
Ethnicity
```

Out[110]:

	ethnicity	number
0	group A	89
1	group B	190
2	group C	319
3	group D	262
4	group E	140

In [111]:

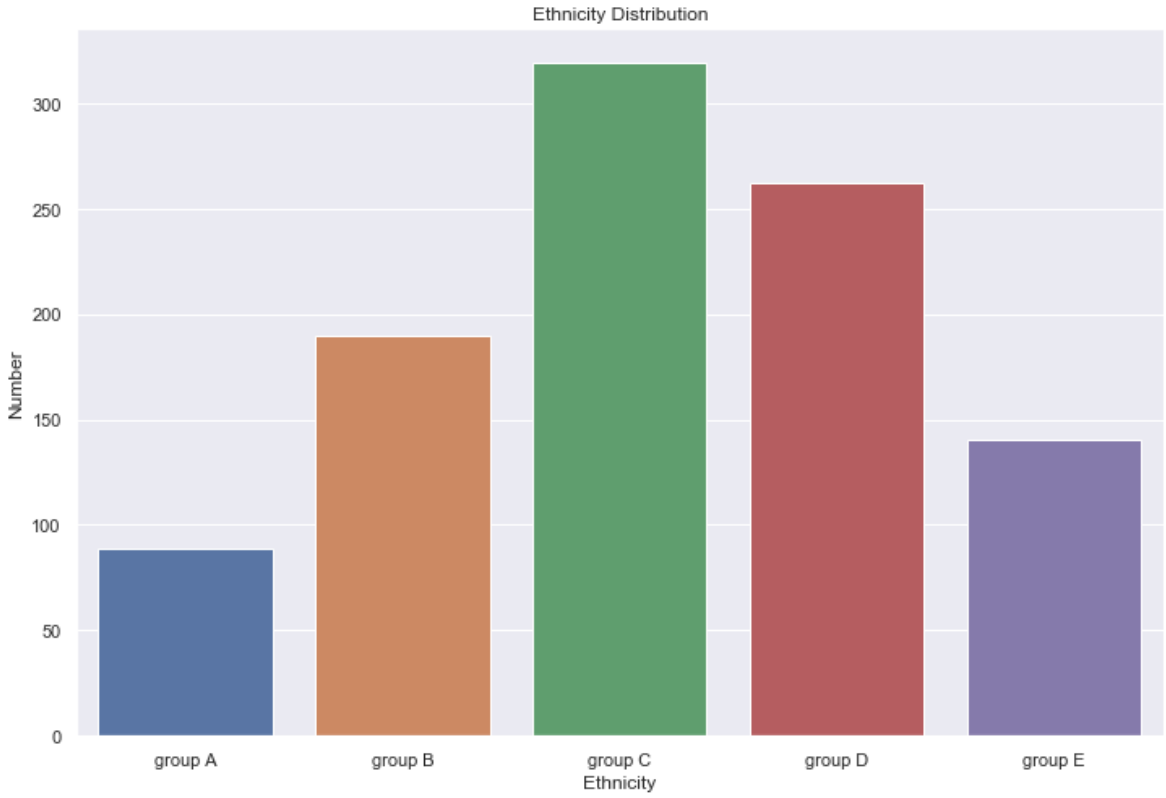
```
race = studentperformance.groupby(['ethnicity'])
Ethnicity = race[['lunch']].count()
Ethnicity.rename(columns={'lunch':'percentage'},inplace=True)
percentage=Ethnicity.apply(lambda x:100* x/x.sum()).reset_index()
percentage
```

Out[111]:

	ethnicity	percentage
0	group A	8.9
1	group B	19.0
2	group C	31.9
3	group D	26.2
4	group E	14.0

In [112]:

```
Ethnicity = studentperformance.groupby('ethnicity').count()
sns.barplot(x=Ethnicity.index, y='lunch', data=Ethnicity).set_title('Ethnicity Distributior
plt.xlabel('Ethnicity')
plt.ylabel('Number')
plt.show()
```



6. Find and plot gender of the students against their ethnicity.

In [113]:

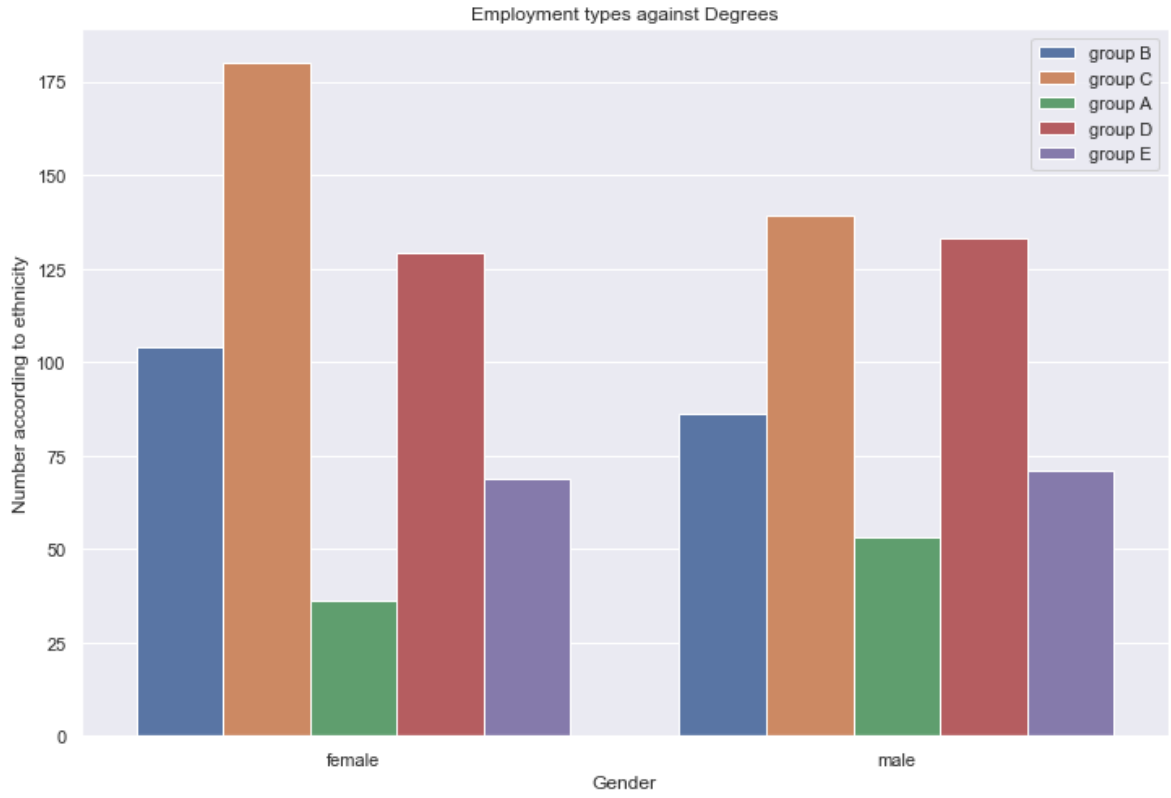
```
Gender_Ethinicity = studentperformance[(studentperformance['gender'] == 'female') | (studer
Gender_Ethinicity.groupby(['gender','ethnicity'])['lunch'].count()
```

Out[113]:

```
gender  ethnicity
female  group A       36
        group B      104
        group C      180
        group D      129
        group E       69
male    group A       53
        group B       86
        group C      139
        group D      133
        group E       71
Name: lunch, dtype: int64
```

In [114]:

```
sns.countplot(x='gender', hue='ethnicity', data=studentperformance).set_title('Employment t
plt.ylabel('Number according to ethnicity')
plt.xlabel('Gender')
plt.legend(loc='upper right')
plt.show()
```



6. Find and plot gender of the students against their parental education.

In [115]:

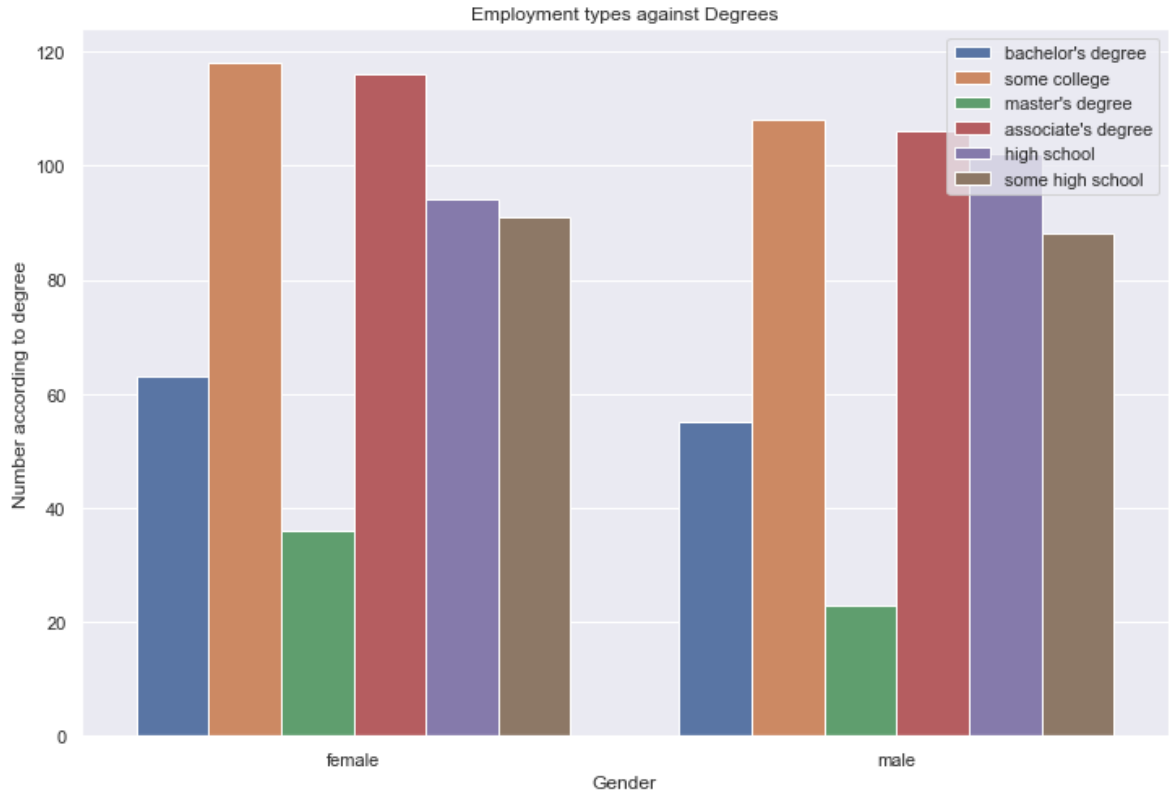
```
Gender_PE = studentperformance[(studentperformance['gender'] == 'female') | (studentperform
Gender_PE.groupby(['gender','parentalEducation'])['lunch'].count()
```

Out[115]:

```
gender  parentalEducation
female  associate's degree    116
        bachelor's degree      63
        high school            94
        master's degree        36
        some college          118
        some high school       91
male    associate's degree    106
        bachelor's degree      55
        high school           102
        master's degree        23
        some college          108
        some high school       88
Name: lunch, dtype: int64
```

In [116]:

```
sns.countplot(x='gender', hue='parentalEducation', data=studentperformance).set_title('Empl
plt.ylabel('Number according to degree')
plt.xlabel('Gender')
plt.legend(loc='upper right')
plt.show()
```



7. Find percentage of students according to lunch type.

In [117]:

```
lunch = studentperformance.groupby(['lunch'])
Lunch = lunch[['gender']].count()
Lunch.rename(columns={'gender':'percentage'},inplace=True)
Percentage=Lunch.apply(lambda x:100* x/x.sum()).reset_index()
round(Percentage,2)
```

Out[117]:

	lunch	percentage
0	free/reduced	35.5
1	standard	64.5

8. Find and plot gender of the students against lunch type.

In [118]:

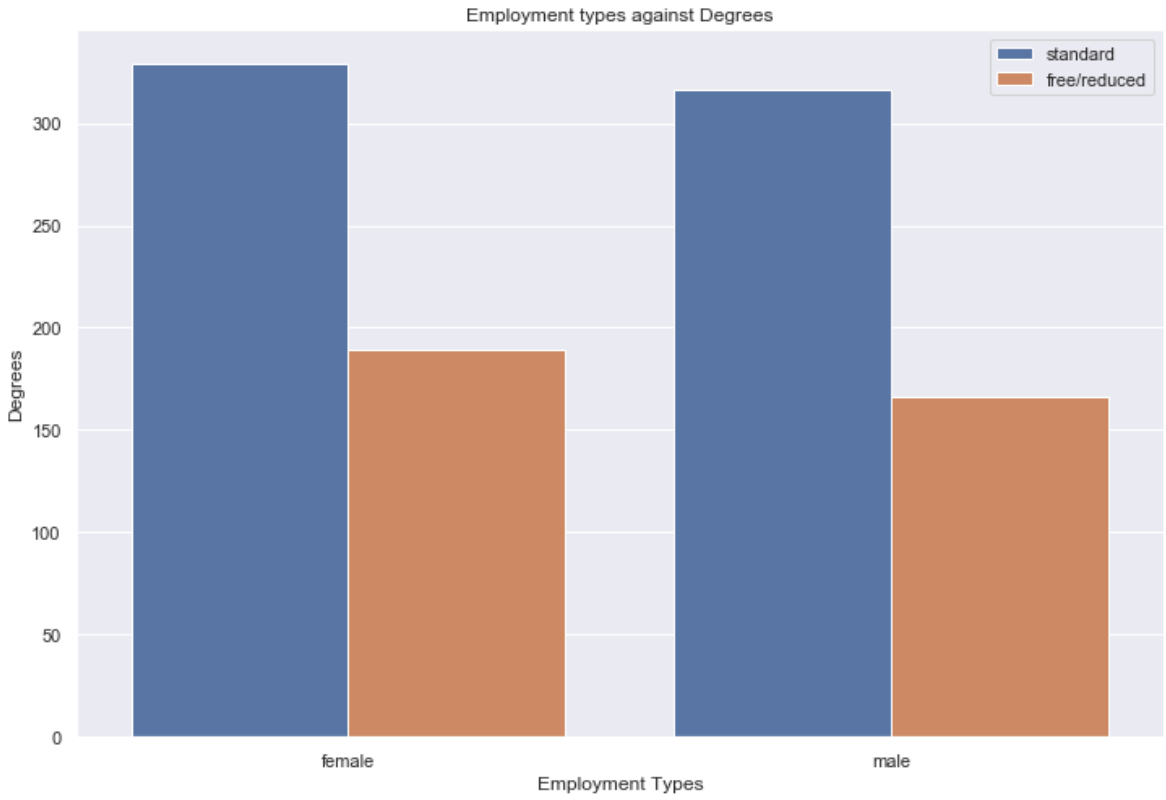
```
Gender_Lunch = studentperformance[(studentperformance['gender'] == 'female') | (studentperf
Gender_Lunch.groupby(['gender','lunch'])['parentalEducation'].count()
```

Out[118]:

```
gender  lunch
female  free/reduced    189
        standard        329
male    free/reduced    166
        standard        316
Name: parentalEducation, dtype: int64
```

In [119]:

```
sns.countplot(x='gender', hue='lunch', data=studentperformance).set_title('Employment types
plt.ylabel('Degrees')
plt.xlabel('Employment Types')
plt.legend(loc='upper right')
plt.show()
```



9. Find percentage of the students according to test preparation.

In [120]:

```
preparation = studentperformance.groupby(['TestPreparation'])
Prep = preparation[['lunch']].count()
Prep.rename(columns={'lunch':'percentage'},inplace=True)
Percentage=Prep.apply(lambda x:100* x/x.sum()).reset_index()
round(Percentage,2)
```

Out[120]:

	TestPreparation	percentage
0	completed	35.8
1	none	64.2

10. Describe the marks of all students according to subjects.

In [121]:

```
round(studentperformance.describe())
```

Out[121]:

	MathScore	ReadingScore	WritingScore
count	1000.0	1000.0	1000.0
mean	66.0	69.0	68.0
std	15.0	15.0	15.0
min	0.0	17.0	10.0
25%	57.0	59.0	58.0
50%	66.0	70.0	69.0
75%	77.0	79.0	79.0
max	100.0	100.0	100.0

11.Find the number of students who passed and failed in Math.

In [122]:

```
PassMark=50
P=studentperformance[(studentperformance['MathScore'] >= PassMark)].shape
F=studentperformance[(studentperformance['MathScore'] < PassMark)].shape
print("Passed in Maths",P)
print("Failed in Maths",F)
```

```
Passed in Maths (865, 8)
Failed in Maths (135, 8)
```

11.Find the number of students who passed and failed in Reading.

In [123]:

```
P=studentperformance[(studentperformance['ReadingScore'] >= PassMark)].shape
F=studentperformance[(studentperformance['ReadingScore'] < PassMark)].shape
print("Passed in Reading",P)
print("Failed in Reading",F)
```

```
Passed in Reading (910, 8)
Failed in Reading (90, 8)
```

11.Find the number of students who passed and failed in Writing.

In [124]:

```
P=studentperformance[(studentperformance['WritingScore'] >= PassMark)].shape
F=studentperformance[(studentperformance['WritingScore'] < PassMark)].shape
print("Passed in Writing",P)
print("Failed in Writing",F)
```

```
Passed in Writing (886, 8)
Failed in Writing (114, 8)
```

12. Create a boxplot to show the average marks of the students and Calculate the five descriptive statistics as shown on the boxplot

In [125]:

```
studentperformance['AvgMarks'] = (studentperformance['MathScore']+studentperformance['Readi
```

In [126]:

```
sns.boxplot(y=studentperformance['AvgMarks']).set_title('Age of Participants')
plt.ylabel('Participants Age')
plt.show()
```

In [127]:

```
mean = int(studentperformance['AvgMarks'].mean())
median = int(studentperformance['AvgMarks'].quantile(0.5))
q1 = int(studentperformance['AvgMarks'].quantile(0.25))
q3 = int(studentperformance['AvgMarks'].quantile(0.75))
minimum = int(studentperformance['AvgMarks'].min())
maximum = int(studentperformance['AvgMarks'].max())

print('Mean: ' + str(mean))
print('Minimum: ' + str(minimum))
print('First Quartile: ' + str(q1))
print('Median: ' + str(median))
print('Third Quartile: ' + str(q3))
print('Maximum: ' + str(maximum))
```

```
Mean: 67
Minimum: 9
First Quartile: 58
Median: 68
Third Quartile: 77
Maximum: 100
```

13.Plot the MathScore and WritingScore of the students on a scatterplot.

In [128]:

```
sns.scatterplot(x='MathScore',y='WritingScore',hue='AvgMarks',data=studentperformance).set_
plt.xlabel("Math Score")
plt.ylabel("Writing Score")
plt.show()
```

14.Plot the MathScore and ReadingScore of the students on a scatterplot.

In [129]:

```
sns.scatterplot(x='MathScore',y='ReadingScore',hue='AvgMarks',data=studentperformance).set_
plt.xlabel("Math Score")
plt.ylabel("Reading Score")
plt.show()
```

15. Create a linear regression between MathScore and AvgMarks

In [130]:

```
from scipy.stats import linregress
slope, intercept, r_value, p_value, std_err = linregress(studentperformance['MathScore'],st
line = [slope*xi + intercept for xi in studentperformance['MathScore']]
plt.plot(studentperformance['MathScore'],line,'r-', linewidth=3)
sns.regplot(studentperformance['MathScore'], studentperformance['AvgMarks']).set_title('Mat
plt.show()
```