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: Manish Heera

Student number: 29833604

Tutorial Day and Time: Wednesday 10 - 12

Tutor: Zhinoos Razavi Hesabi

Environment: Python 3.6 (32 bit) and *Distribution (Anaconda 5.3.0 (32-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 [48]:
```
#loading libraries
import numpy as np    #python extension module, used to provide fast and efficient operation on homogenous data type
import pandas as pd   # importing the library pandas and reference it as pd
                      # pandas are software programming library used in python for data manupulationa and analysis
import seaborn as sns # importing the library seaborn and reference it as sns, used for statistical graphics in python
import matplotlib.pyplot as plt #importing the plotting module matplot and referncing it as plt

#to view the plots in the jupyter notebook itself using matplotlib
%matplotlib inline
```

1. Demographic Analysis

So what does a data scientist look like?

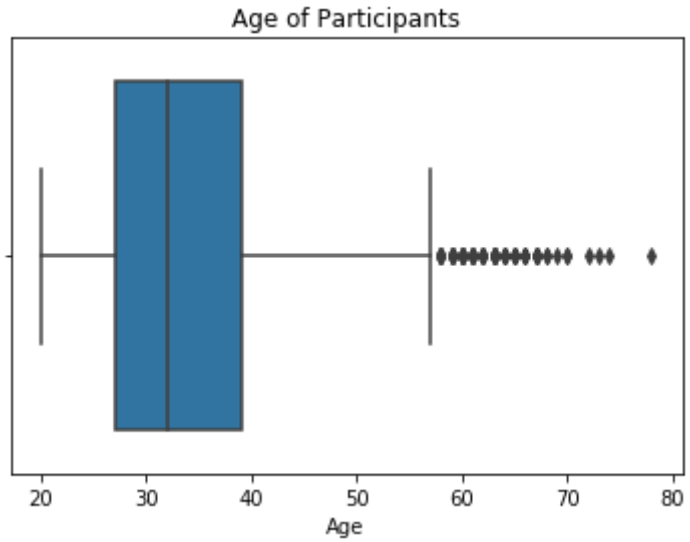
Let's get a general understanding of the characteristics of the survey participants. Demographic overviews are a standard way to start an exploration of survey data. The types of participants can heavily affect the survey responses.

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 [5]:
```
response = pd.read_csv('multipleChoiceResponses.csv', sep=',') #reading the multipleChioceResponses.csv
#file into pandas dataframe

#boxplot, where x axis represents Age, and set_title function is used to label the plot
plot = sns.boxplot(x = response ['Age']).set_title('Age of Participants')
```



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 [6]:
```
#minimum outliners :
#maximum outliners :
#median            :
#first quartile (the middle number between the smallest number (not the “minimum”) and the median of the dataset)
#third quartile (the middle value between the median and the highest value (not the “maximum”) of the dataset)

response['Age'].median()  #calculating median: 32.0
response['Age'].max()    #calculating max value: 78
response['Age'].min()    #calculating min value: 20
meanAge = response['Age'].mean()  #calcultaing the mean of the age column and storing result in meanAge: 34.2714
round(meanAge)  # rounds off the number to the nearest integer using round funtion
```

Out[6]: 34

Answer

It can be depicted from the above calculation that: WE have median vlaue as 32 max value: 78 min value: 20 we are using round function to round about the value for the mean Age

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

Following information can be depicted from the box plot:

1. Median is 32, which is the middle value of our dataset
2. First quartile is about 27
3. Third quartile is almost 40 (39 approx)
4. Maximum Age is 78, can be seen in the plot
5. Minimum Age is 20

6. Outlier, observation point distant from other observation, in above boxplot i have got few outliners the extreme one has the value same as maximum value for Age.

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 [7]:
```
e = response[['Age','CurrentJobTitleSelect']] # getting age and job column and storing it into e variable

#applying lambda function to count the age under 18 and over 65
age_dataScientist = {'Age':{'Under 18': lambda x: sum(e<18 for e in x ) , 'Over 65': lambda y: sum(e>65 for e in y)  }}
groupbyJob = e.groupby('CurrentJobTitleSelect').agg(age_dataScientist).reset_index()
groupbyJob.columns = groupbyJob.columns.droplevel(0)        # drop the top level in column hierarchy
groupbyJob.rename(columns = {'':'CurrentJobTitleSelect'},inplace = True) #rena,ing the column
groupbyJob
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[7]:

	CurrentJobTitleSelect	Under 18	Over 65
0	Business Analyst	0	2
1	Computer Scientist	0	1
2	DBA/Database Engineer	0	1
3	Data Analyst	0	0
4	Data Miner	0	0
5	Data Scientist	0	4
6	Engineer	0	2
7	Machine Learning Engineer	0	1
8	Operations Research Practitioner	0	0
9	Other	0	2
10	Predictive Modeler	0	0
11	Programmer	0	0
12	Researcher	0	2
13	Scientist/Researcher	0	2
14	Software Developer/Software Engineer	0	2
15	Statistician	0	0

Answer

19 Respondands were over 65 years old where as, none were under 18

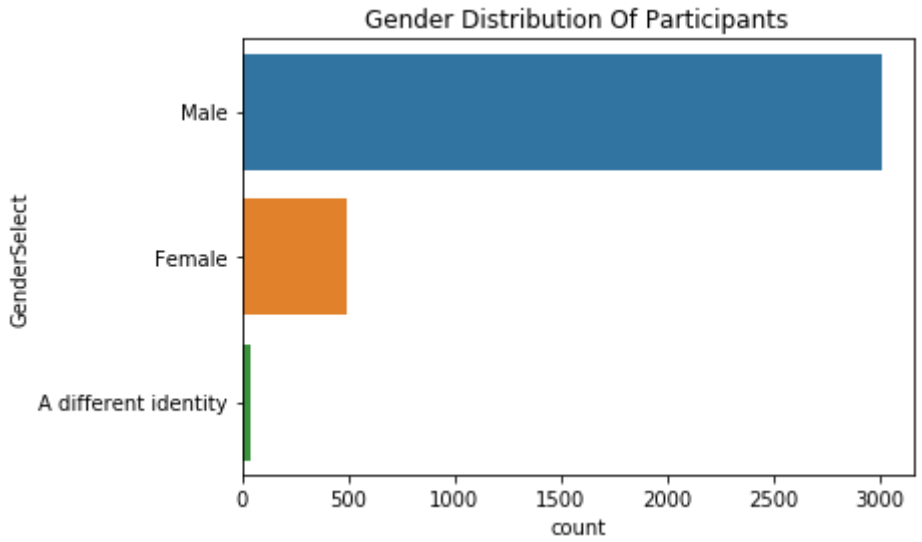
1.2 Gender

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 [8]:
```
#using countplot of seaborn to show the count of obseravtion

plt.title('Gender Distribution Of Participants') #Setting the title of the plot
sns.countplot(data = response, y = 'GenderSelect') #y axis as gender distribution at x axis we have response of the respon
```

Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x153414f6a58>



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

In [9]:
```
# Your code

#counting number of men
male = response['GenderSelect'] == "Male" #storing boolean series in male, only Men will be stored
response["new"] = male #inserting new series in response dataframe
Men = response[male]  #storing result for female response in Men
Men['GenderSelect'].count() #selecting GenderSelect column from Men dataframe and counting the same

#counting number of female
female = response['GenderSelect'] == "Female" #storing boolean series in female
response["new"] = female #inserting new series in response dataframe
Women = response[female] #storing result for female response in Women
Women['GenderSelect'].count()

#Adding the count for men and women and storing it into c
c = Men['GenderSelect'].count() + Women['GenderSelect'].count()

percentageMen = (Men['GenderSelect'].count() / c) * 100 #claculating the percentage of Men
percentageMen # percentage of Men were 85.84
percentageWomen = (Women['GenderSelect'].count() / c) * 100 #calculating the percentage of Women
percentageWomen  #percentage of Women is 14.155
```

Out[9]:
```
14.15525114155251
```

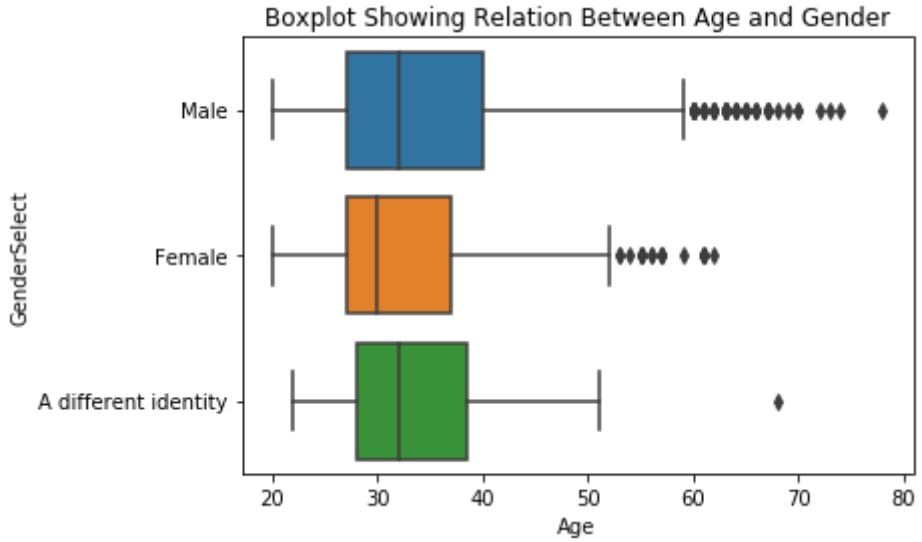
Answer Percentage of men were: 85.84

```
Percentage of women were: 14.155
```

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 [10]:
```
#creating a boxplot with x axis as Age and y as Gender
#x axis storing Age coulmn and y axis storing Gender
#Using set_title to give title to the plot
AgeVsGenderPlot = sns.boxplot(x = response ['Age'], #using seaborn libraray for plotting
                              y = response['GenderSelect']).set_title('Boxplot Showing Relation Between Age and Gender')
```



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 [11]:
```
#As per boxplot we'll get the mean max and min values for the age of all three Gender catagories
fun = {'GenderSelect':{'GenderSelect':'count'},'Age':{'Average Age':'mean','Oldest':'max','Youngest':'min'}}
groupbyGender = response.groupby('GenderSelect').agg(fun).reset_index() #turning GenderSelect into column value
groupbyGender
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[11]:

		GenderSelect	Age		
		GenderSelect	Average Age	Oldest	Youngest
0	A different identity	36	34.666667	68	22
1	Female	496	32.735887	62	20
2	Male	3008	34.637633	78	20

Answer

Amount three Gender catgories:

Male has the maximum respondednts with oldest respondant having age as 78

Youngest Male and Female is 20 years old

1.3 Country

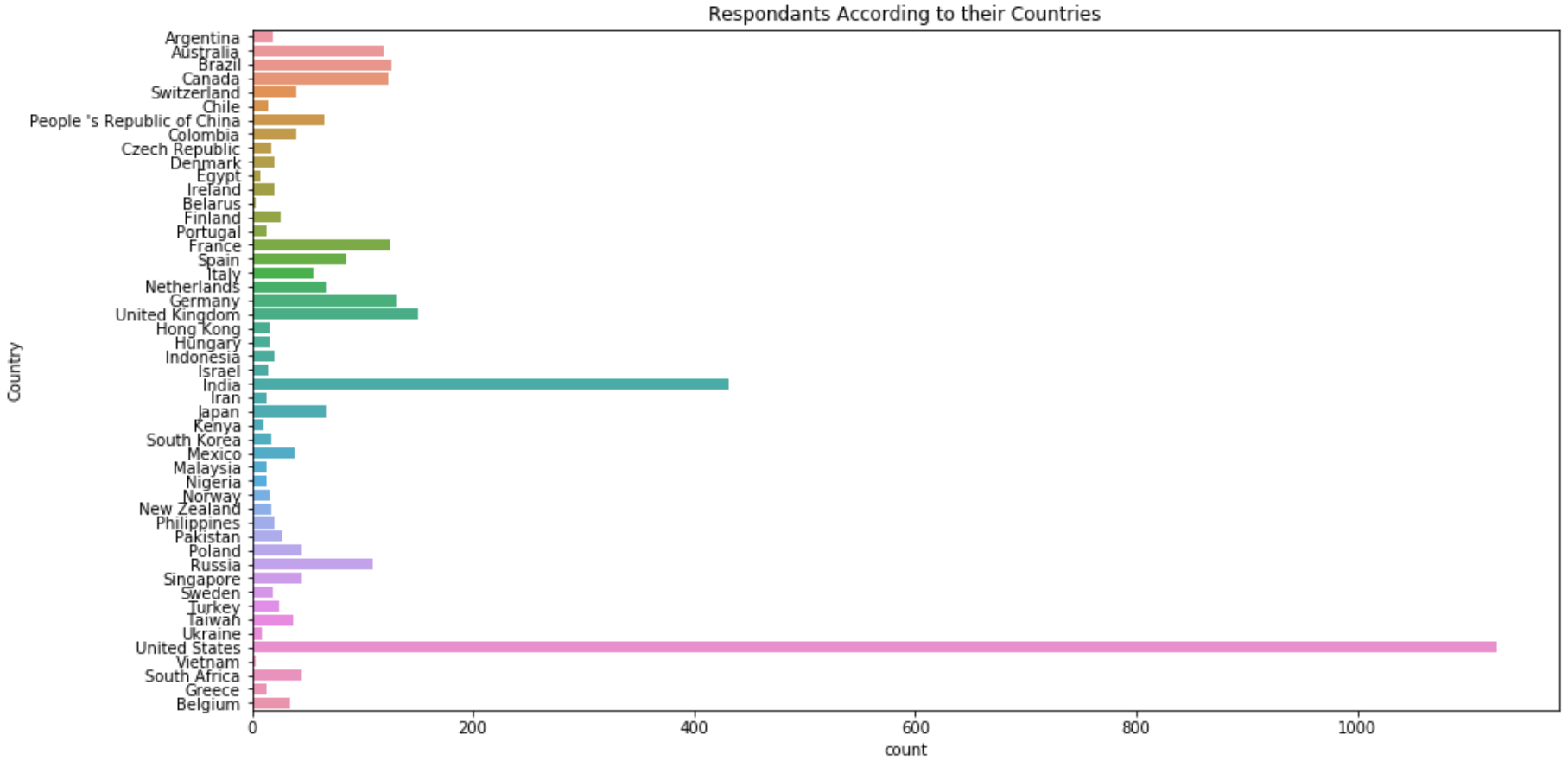
We know that people practise data science all over the world. The United States is thought of as a 'hub' of commercial data science as well as research followed by the United Kingdom and Germany.

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 [12]:
```
d = response["Country"] #selecting the column with country name assigning it to d
plt.figure(figsize = (15,8)) #using figure function to fix the size of the plot
#setting the title for the bar plot
plt.title('Respondants According to their Countries')
#A count plot can be thought of as a histogram across a categorical, instead of quantitative, variable.
#Showing count value all countries
sns.countplot(data = response, y = 'Country')
pd.value_counts(d)[:5] #top five country respondants
TotalC = d.count() #count is respondants is 3540
TopFC = pd.value_counts(d)[:5].sum() #sum of topfive respondants == 1965
percentage = (TopFC / TotalC) * 100 #55.5

print("Percentage of respondands from top 5 countries : ",percentage) #printing the percentage
```

```
Percentage of respondands from top 5 countries :  55.50847457627118
```



Answer

Top five respondants: COUNTRY COUNT United States 1126 India 431 United Kingdom 151 Germany 130 Brazil 127

Total count = 3540 (For all respondants) percentage of top 5 countries = 55.5%

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

United states have the maximum reponded and UK falls third in that category followed by Germany with count 1126 151 and 130 respectively

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 [13]:
```
#using the aggregate function mean and median to calculate mean and median of Age column
fun_age = {'Age':{'Mean': 'mean','Median': 'median'}}
#slicing based upon loc of the countries
response.groupby(['Country','GenderSelect']).agg(fun_age).loc[['Australia','Germany','India',
                                                               'United Kingdom','United States'],:]
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[13]:

		Age	
		Mean	Median
Country	GenderSelect		
Australia	Female	35.000000	34.0
	Male	37.158416	36.0
Germany	Female	31.428571	29.0
	Male	36.629310	34.0
India	A different identity	22.000000	22.0
	Female	29.061224	28.0
	Male	29.553806	28.0
United Kingdom	A different identity	36.000000	36.0
	Female	33.636364	33.0
	Male	35.811024	33.0
United States	A different identity	38.727273	43.0
	Female	34.370892	31.0
	Male	36.906874	34.0

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

Answer The mean age of all the countries except INDIA lies in 'thirties' and mean and median age of Males are greater than females. Except Australia all other countries have 'A different Identity' Gender identifed.

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.

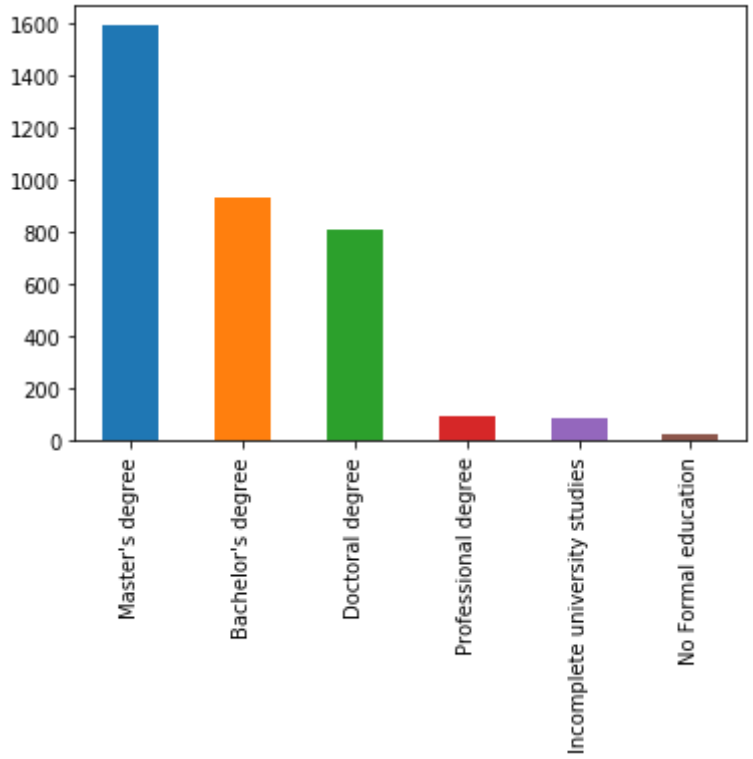
14. Plot and display as text output the number and percentage of respondents with each type of formal education.

In [14]:
```
#storing FormalEducation column in Edu reference
Edu = response['FormalEducation']

#number of reponsdent with each type of formal education
pd.value_counts(Edu)

#using value_counts to obtain unique obsevation for formalEducation and ploting it as 'bar' plot
pd.value_counts(Edu).plot.bar()


#TEXT OUTPUT
TextOutput = pd.value_counts(Edu) #text count of number of respondands as per formal Education
print("Text Output is : \n",TextOutput) #text output \n used for new line

#PERCENTAGE OF RESPONDENTS
Edu.count()


function = {'FormalEducation':{'Count for Degree type': lambda x: sum(e is not None for e in x)}}
Data = response.groupby('FormalEducation').agg(function).reset_index() #resetting the column
Data.columns = Data.columns.droplevel(0) #dropping the coulmn level
Data.rename(columns ={'':'FormalEducation'}, inplace = True) #renaming first column as FormalEducation
#Data #and hence we got the data for respondants as per formal education

Data['Total Count'] = 3540 #total respondants
Data # here we have the required dataset
Data['Percentage'] = Data['Count for Degree type'] / Data['Total Count'] * 100 #calculating the percentage
Data
```

```
Text Output is :
 Master's degree                1594
Bachelor's degree               930
Doctoral degree                 808
Professional degree              96
Incomplete university studies    87
No Formal education              25
Name: FormalEducation, dtype: int64

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[14]:

	FormalEducation	Count for Degree type	Total Count	Percentage
0	Bachelor's degree	930	3540	26.271186
1	Doctoral degree	808	3540	22.824859
2	Incomplete university studies	87	3540	2.457627
3	Master's degree	1594	3540	45.028249
4	No Formal education	25	3540	0.706215
5	Professional degree	96	3540	2.711864



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

yes!

- No job advertiser would like to go for higher degree graduate if the same requirement can be filled with someone holding a bachelor's degree
- I believe Doctoral Degree will be suited to someone who wishes to do more R&N in his or her field of specialisation.

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 [15]:

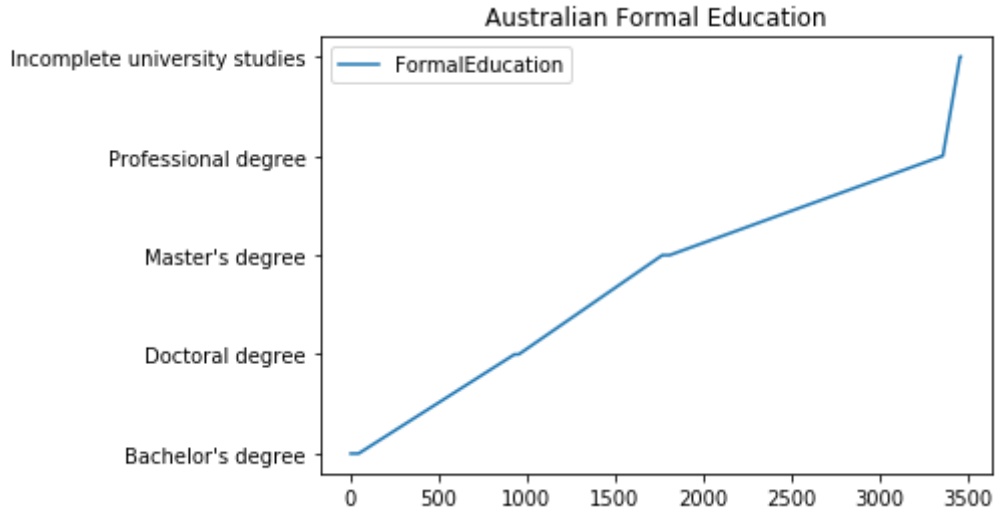
```
Nation = response['Country'] == 'Australia' #Filtering based on country i.e. Australia
response["new"] = Nation #inserting new series in response dataframe
country = response[Nation] #storing result for country
country.FormalEducation# display data for Australia
plt.plot(country.FormalEducation) # plot for Australia as per formal Education
plt.title('Australian Formal Education') #setting title for the graph
plt.legend()

#Text Output
Education = response['FormalEducation']
text = pd.value_counts(Education) #using value_counts to display each degree type with count

#Percentage

#filtering out Data as per country == Australia
nation = response['Country'] == 'Australia'
response['new'] = nation
AussieData = response[nation]
fun ={'FormalEducation':{'Aussie_Count': lambda x: sum(e is not None for e in x)}}
#group by formal education
AussieData = AussieData.groupby('FormalEducation').agg(fun).reset_index()
AussieData.columns = AussieData.columns.droplevel(0)
AussieData.rename(columns ={'':'FormalEducation'}, inplace = True) #renaming first column as FormalEducation
#AussieData #hence we obtained the data for count of Australians with their degree type


funn = {'FormalEducation':{'Overall_count': lambda x: sum(e is not None for e in x)}}
AussieData2 = response.groupby('FormalEducation').agg(funn).reset_index()
AussieData2.columns = AussieData2.columns.droplevel(0)
AussieData2.rename(columns={'':'FormalEducation'}, inplace = True)
#AussieData2 #hence we obtained the over-all count for formal education

#merging the two datasets i.e. AussieData and AussieData2
DataSet = pd.merge(AussieData, AussieData2, on = 'FormalEducation', how = 'left') #sucessfully merged the two datasets bas
DataSet['Percentage'] = AussieData['Aussie_Count']/AussieData2['Overall_count'] * 100 #calculating the percentage of Aussi
print(DataSet,"\n\n","TEXT OUTPUT\n",text) # percentage and textoutput of Australia
```

```
                FormalEducation  Aussie_Count  Overall_count  Percentage
0             Bachelor's degree            45            930    4.838710
```

```
1                Doctoral degree           25            808    3.094059
2  Incomplete university studies            5             87    5.747126
3                Master's degree           42           1594    2.634881
4             Professional degree           2             96    8.000000
```

```
 TEXT OUTPUT
 Master's degree                 1594
Bachelor's degree                930
Doctoral degree                  808
Professional degree               96
Incomplete university studies     87
No Formal education               25
Name: FormalEducation, dtype: int64
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```



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

In [16]:
```
#aggregate function Degree to calculate mean and median of Age
Degree = {'Age':{'Average Age':'mean', 'Median Age': 'median'}}
#grouping them by degree type
groupbyEdu = response.groupby('FormalEducation').agg(Degree).reset_index() #performing
groupbyEdu.columns = groupbyEdu.columns.droplevel(0)
groupbyEdu.rename(columns ={'' :'Degree Type'}, inplace = True)#renaming tthe first column after performing droplevel to z
groupbyEdu #to display output
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[16]:

	Degree Type	Average Age	Median Age
0	Bachelor's degree	30.632258	28.0
1	Doctoral degree	39.235149	37.0
2	Incomplete university studies	36.011494	35.0
3	Master's degree	33.746550	31.0
4	No Formal education	41.680000	42.0
5	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.

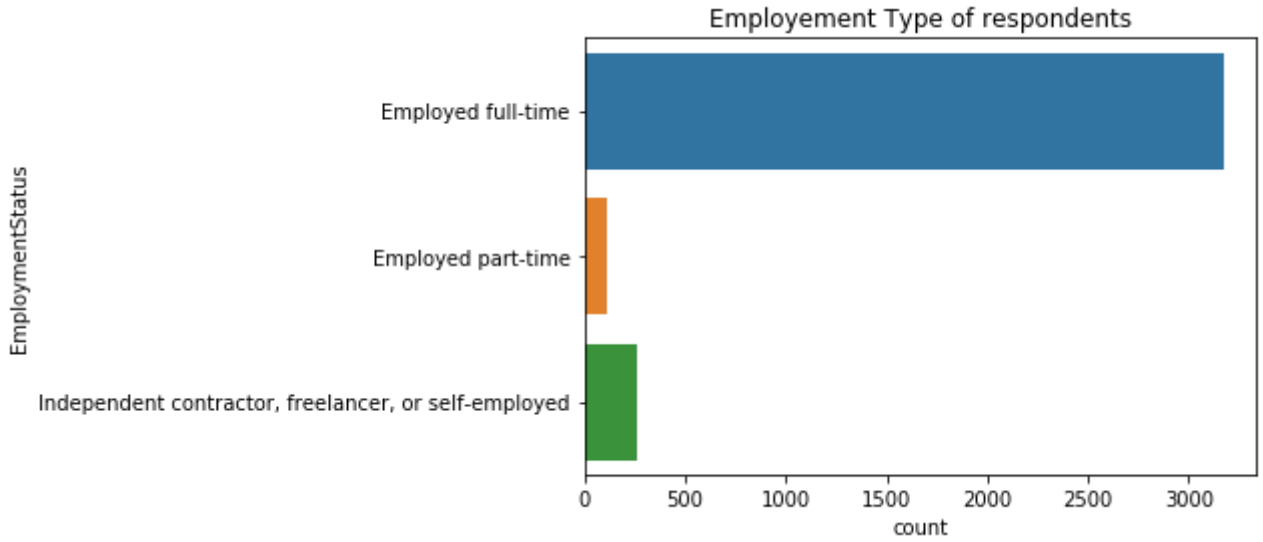
3.1 Employment status

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 [17]:
```
Work = response['EmploymentStatus']
#for the title of graph
plt.title('Employement Type of respondents')
#count plot using sns refernce of seaborn to plot type of employement
sns.countplot(data = response, y = 'EmploymentStatus')
```

Out[17]: <matplotlib.axes._subplots.AxesSubplot at 0x1534187b048>

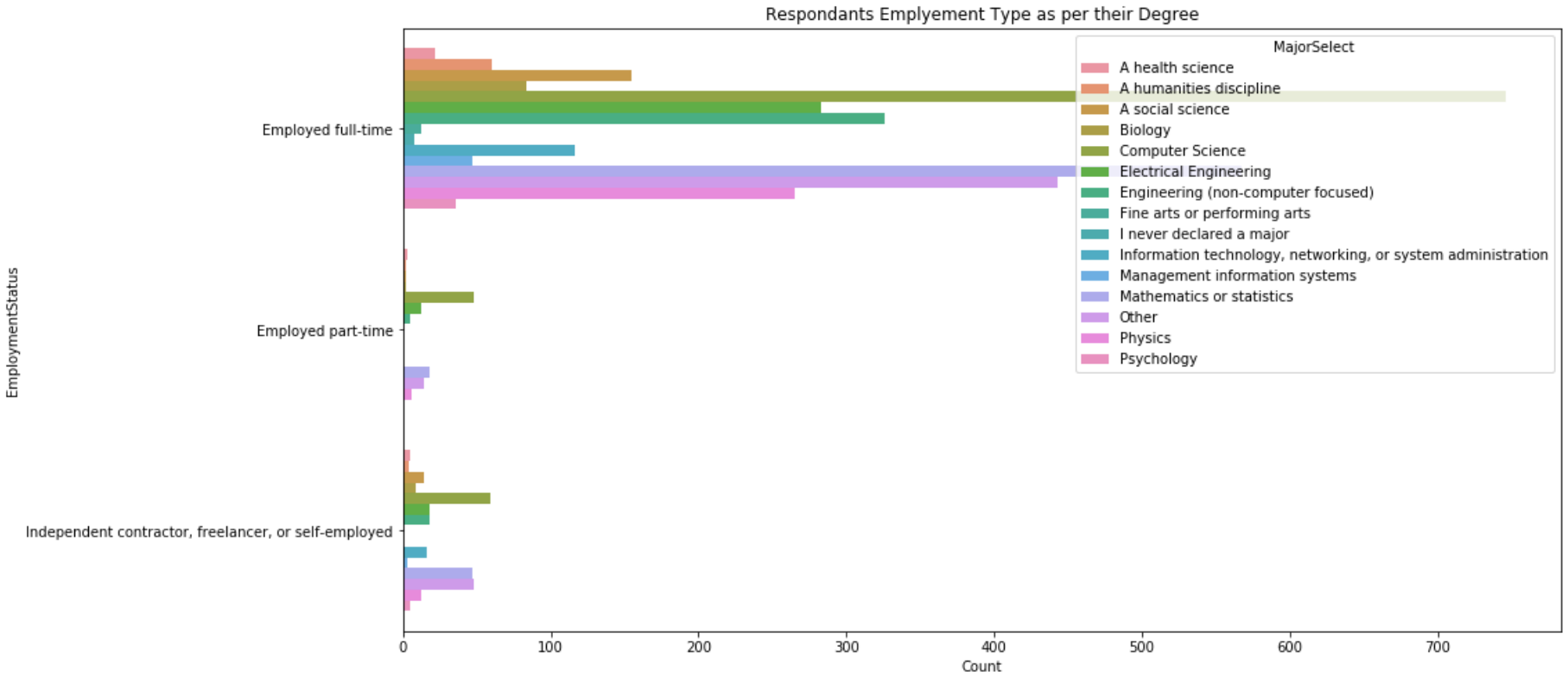


19. You may be wondering if your own degree and experince will help you gain full time employment after you graduate.

Plot the respondants employment types against their degrees.

In [20]:
```
#grouping by EmplyementStatus and MajorSelect

groupbyEmpMajor = response.groupby(['EmploymentStatus','MajorSelect']).count()
groupbyEmpMajor = groupbyEmpMajor.reset_index()
groupbyEmpMajor.rename(columns = {'Age':'Count'}, inplace = True)

plt.figure(figsize = (15,8))
sns.barplot(x = 'Count', y = 'EmploymentStatus', hue = 'MajorSelect', data = groupbyEmpMajor)
plt.title('Respondants Emplyement Type as per their Degree')
```

Out[20]: Text(0.5,1,'Respondants Emplyement Type as per their Degree')



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

As per the data Studying Computer Science can leadd to full time employement, maximum people have got full time job in this MajorSelect.

As far as IT, networking or system administration is concerned there is no part time jobs for them, and very minimum chances of getting contract based work as well.

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 [22]:

```
EmpCountry = ((response.Country == 'Australia') | (response.Country == 'United States') | (response.Country == 'United Kin

EC = response[EmpCountry] #here we are storing data for Aus/UK/US only
#EC
Employed = EC.EmploymentStatus == 'Employed full-time' #filtering out AUS/UK/USA as per Full Time Employement
dc = EC[Employed] #and here we have the required dataset consiting of AUS/UK/USA as per Full time employement

#Employed
#dc = response[Employed]
#dc


func = {'EmploymentStatus':{'Count for Full-timers AUS/UK/US': lambda x: sum(e is not None for e in x)}}
DC = dc.groupby('Country').agg(func).reset_index() #numbers of full-timers in Australia/UK/USA
DC.columns = DC.columns.droplevel(0) #leveldrop for the column
DC.rename(columns ={'' :'Country'}, inplace = True) #renaming first Column
#DC #hence we got the full time employed respondednts in UK/US/andAUS

#data for Full-timers and non full timers cumulative data from AUS/UK/USA
fun = {'EmploymentStatus':{'Overall Count': lambda x: sum(e is not None for e in x)}}
ec = EC.groupby('Country').agg(fun).reset_index() #numbers of full-timers in Australia/UK/USA
ec.columns = ec.columns.droplevel(0)
ec.rename(columns ={'' :'Country'}, inplace = True) #renaming first Column
#ec

BC = pd.merge(DC,ec, on = 'Country', how = 'left')
BC['Percentage'] = round(BC['Count for Full-timers AUS/UK/US']/BC['Overall Count'] * 100)
BC.rename(columns = {'Num_x':'ALL JOB TYPES COUNT','Num_y':'FULL-TIME JOB count'}, inplace = True)
print("Percentage\n",BC) #printing out the percentage of full time AUS/UK/USA respondants
```

```
Percentage
           Country  Count for Full-timers AUS/UK/US  Overall Count  Percentage
0       Australia                              101            119        85.0
1  United Kingdom                              137            151        91.0
2   United States                             1030           1126        91.0

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

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

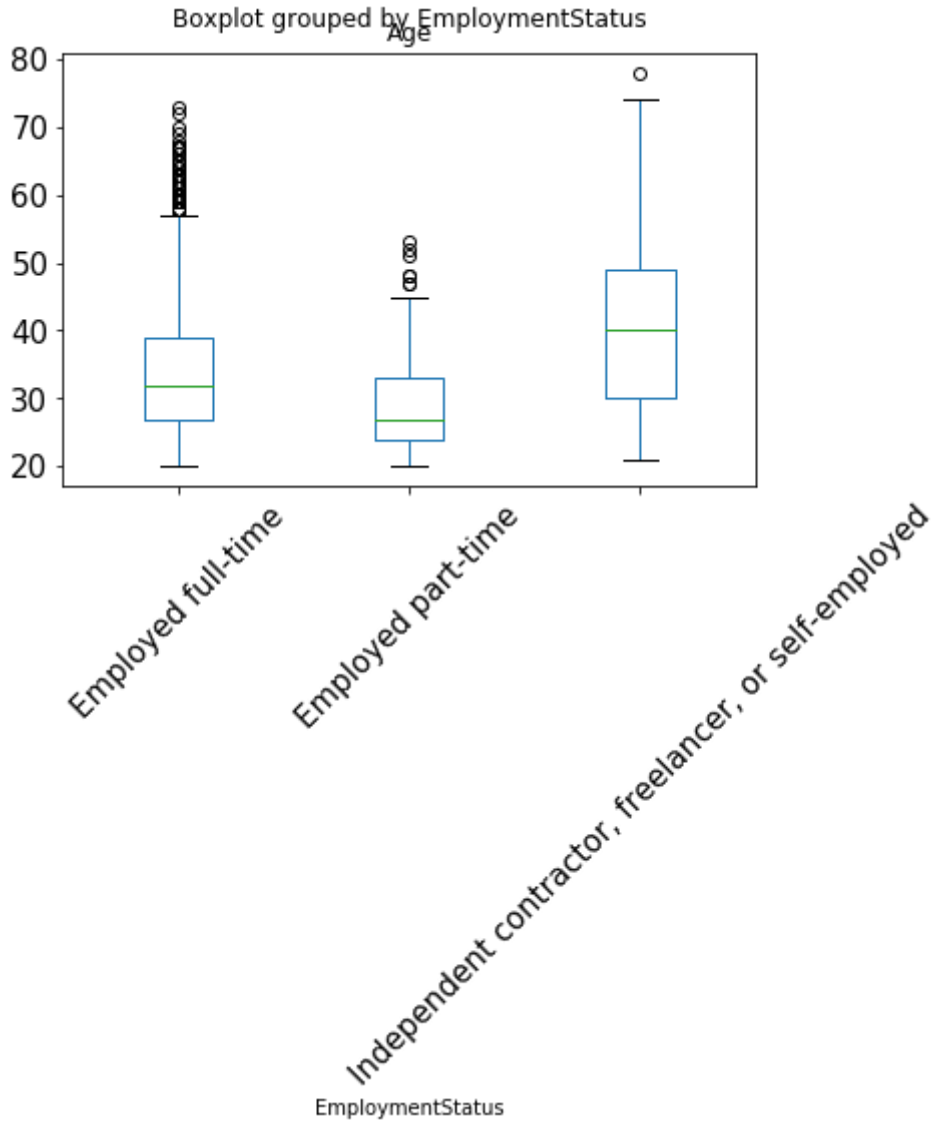
Let's find out the median age of employees by type of employment.

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

In [23]:
```
# Your code
funt = {'Age':{'Oldest':'max', 'Youngest':'min','Median Age': 'median'}}
groupbyEmp = response.groupby('EmploymentStatus').agg(funt).reset_index()
groupbyEmp.columns = groupbyEmp.columns.droplevel(0)
groupbyEmp

#boxplot
response.boxplot(column = 'Age', by = 'EmploymentStatus',grid=False, rot=45, fontsize=15,layout=(1, 1))
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[23]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x15341aa4e48>
```



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 [24]:
```
#importing the conversionRate.csv file

currency = pd.read_csv('conversionRates.csv', sep = ',')

#merging 'multipleChoiceResponse' file which is in dataframe 'response' to currency which is dataframe for 'conversionRate
merging = response.merge(currency, left_on = 'CompensationCurrency', right_on = 'originCountry')

#considering we need all country's currency in AUD
#converting salary data into AUD
merging['AudSal'] = merging.CompensationAmount * merging.exchangeRateAUS
merging

#dropping out unwanted columns
merging.drop(columns = ['Unnamed: 10', 'new'])

#Calculating Maximum and Median Salary
Sal = {'AudSal':{'Maximum Sal':'max','Median Sal':'median'}}

#output for median and max sal in all countries
ac = merging.groupby('Country').agg(Sal)
ac = ac.reset_index() #resetting the indexing to remove multilevel indexing
ac.columns = ac.columns.droplevel(0) #dropping the column level
ac.rename(columns = {'':'Country'},inplace = True)
ac
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[24]:

	Country	Maximum Sal	Median Sal
0	Argentina	72844.661000	26224.077960
1	Australia	500000.000000	120000.000000
2	Belarus	89428.724580	64090.585949
3	Belgium	298095.748600	81678.235116
4	Brazil	200265.483500	36047.787030
5	Canada	307993.668300	92398.100490
6	Chile	72061.920000	30025.800000

	Country	Maximum Sal	Median Sal
7	Colombia	193192.157635	23018.526000
8	Czech Republic	79954.133000	27983.946550
9	Denmark	160267.228000	100167.017500
10	Egypt	21184.828800	8262.083232
11	Finland	149047.874300	59619.149720
12	France	193762.236590	71542.979664
13	Germany	327905.323460	89428.724580
14	Greece	136378.804985	44714.362290
15	Hong Kong	239305.255500	46903.830078
16	Hungary	72839.670000	16898.803440
17	India	447782.032000	15575.027200
18	Indonesia	71044.875000	5257.320750
19	Iran	29913.600000	5982.720000
20	Ireland	223571.811450	96881.118295
21	Israel	169295.558880	88174.770250
22	Italy	372619.685750	64090.585949
23	Japan	227044.400000	79465.540000
24	Kenya	36367.488000	1454.699520
25	Malaysia	59275.342400	30823.178048
26	Mexico	91408.807100	18563.019288
27	Netherlands	372619.685750	95390.639552
28	New Zealand	272139.572100	77106.212095
29	Nigeria	62549.388000	1111.989120
30	Norway	174810.484200	111640.331955
31	Pakistan	49605.763200	6377.883840
32	People 's Republic of China	572098.068000	41953.858320

	Country	Maximum Sal	Median Sal
33	Philippines	47727.375150	15272.760048
34	Poland	105110.493600	29781.306520
35	Portugal	93900.160809	46502.936782
36	Russia	156167.064000	15616.706400
37	Singapore	277669.105500	74045.094800
38	South Africa	172755.667800	57585.222600
39	South Korea	220862.200000	46381.062000
40	Spain	149047.874300	59619.149720
41	Sweden	132646.234050	78027.196500
42	Switzerland	292606.286850	130047.238600
43	Taiwan	124530.417000	33208.111200
44	Turkey	90918.722250	21820.493340
45	Ukraine	40170.407760	14346.574200
46	United Kingdom	790290.000000	87810.000000
47	United States	685520.559350	137104.111870
48	Vietnam	7897.190400	3564.704000

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

Answer No, here we have obtained the salary figures after conversion as per AUDcurrecy rate

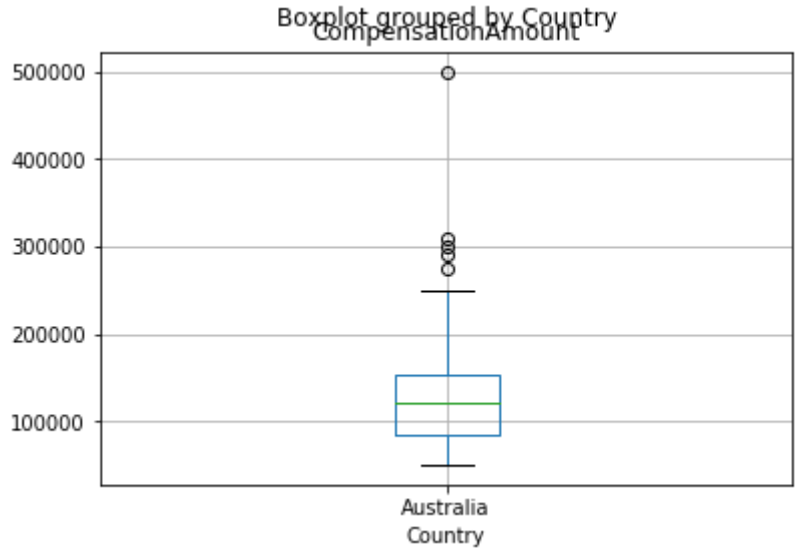
4.2 Salary by country

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 [76]:
```
#box plot for sal of Australians
Aussies = response['Country'] == 'Australia'
#boxplot by taking salary and country
group = response.loc[response['Country'] == 'Australia'].boxplot(column = 'CompensationAmount', by = 'Country')
group

# calculating max and median salaries for Australian respondants

fun = {'CompensationAmount':{'Maximum Salary': 'max', 'Median Of Salary': 'median'}}
output = response.groupby('Country').agg(fun).loc['Australia',:]  #grouping by country, slicing Australia as per loc
print(output) #printing out max and median Salries for Australian
```

```
CompensationAmount  Maximum Salary      500000.0
                    Median Of Salary    120000.0
Name: Australia, dtype: float64

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```



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

Answer

Yes, the value for Max and median Sal for Australia remains the same, as it was in AUD then and Now.

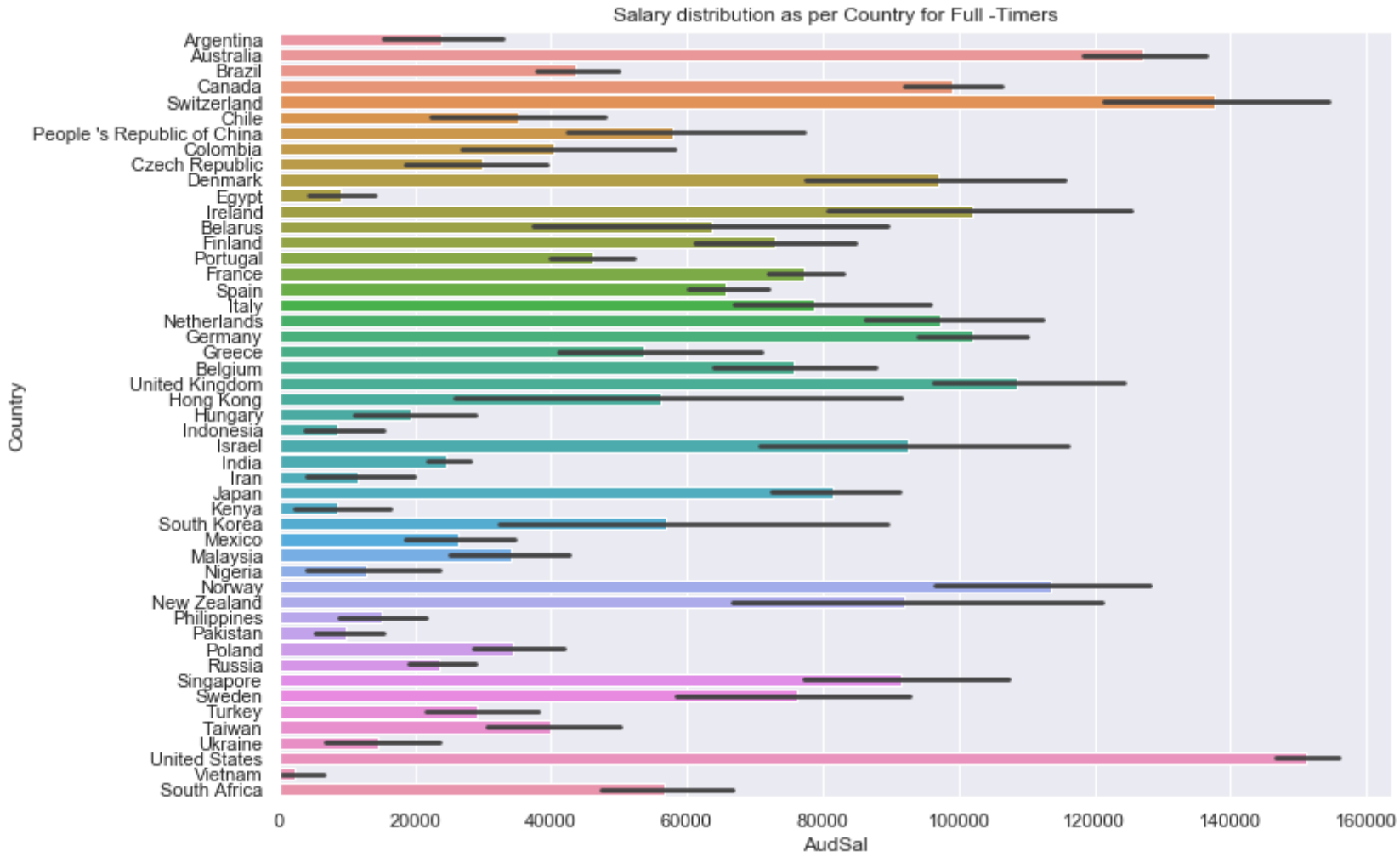
27. Australia's salaries look pretty good.
Plot the salaries of all countries on a bar chart.

Hint: Adjust for full-time employees only

In [131]:
```
#using the merging dataframe which is the combination of both csv's

#copying the data for full timers
FullTimers = merging['EmploymentStatus'] == 'Employed full-time'
merging["new"] = FullTimers #inserting new series in response dataframe
EmployementT = merging[FullTimers] #storing result for EmployementT
country = pd.value_counts(EmployementT['Country'])
#country

#setting the figure size
sns.set(rc={'figure.figsize':(11.7,8.27)})

#plot for fulltimers , country vs salary
sns.barplot(EmployementT['AudSal'], EmployementT['Country'], data = EmployementT)
plt.title('Salary distribution as per Country for Full -Timers')
```

```
C:\ProgramData\Anaconda3\lib\site-packages\scipy\stats\stats.py:1713: FutureWarning: Using a non-tuple sequence for mu
ltidimensional indexing is deprecated; use `arr[tuple(seq)]` instead of `arr[seq]`. In the future this will be interpr
eted as an array index, `arr[np.array(seq)]`, which will result either in an error or a different result.
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```

Out[131]: Text(0.5,1,'Salary distribution as per Country for Full -Timers')



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

Answer The black mark at the top of the bars represent the positive error where as anything beneath the bar is negative error.

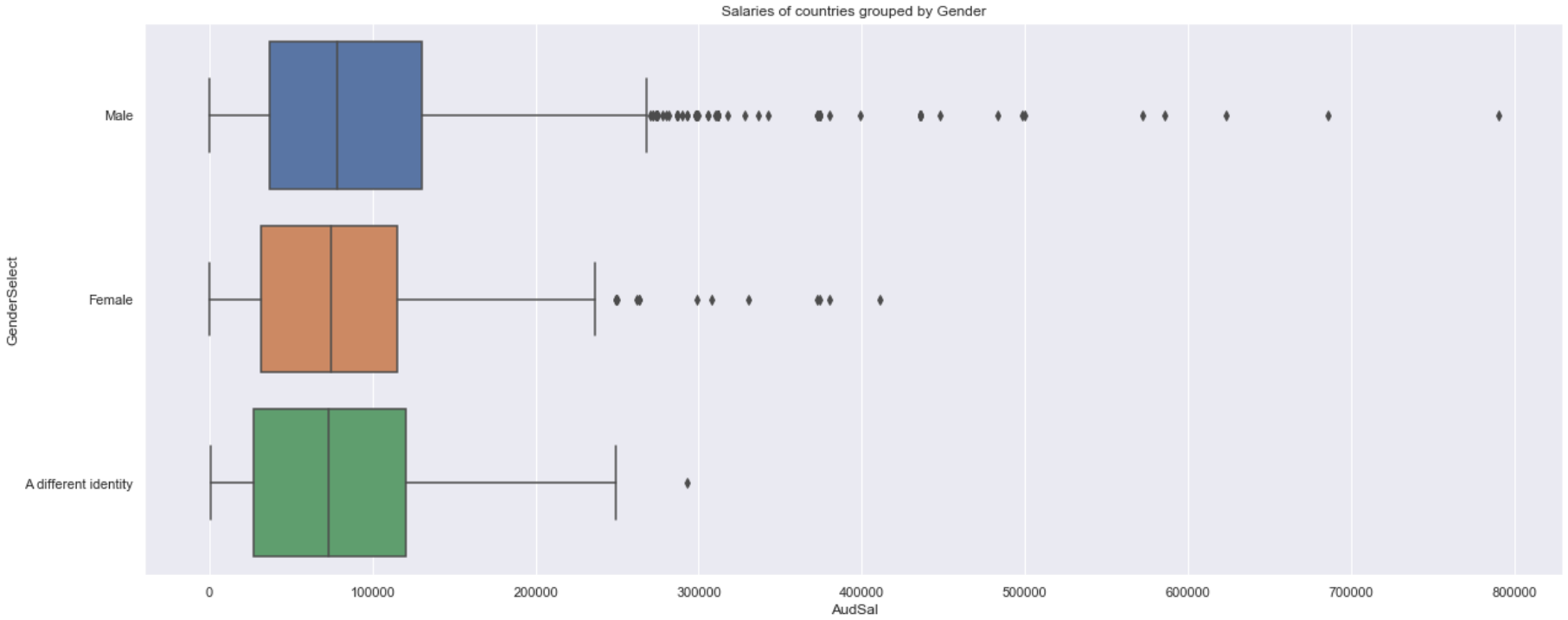
4.3 Salary and Gender

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 [25]:
```
#Yourcode
sns.set(rc={'figure.figsize':(20.7,8.27)})

#using the merging file, that we have obtained after merging both csv's
#ploting the boxplot with x axis as Sal and y as Gender
plt.title('Salaries of countries grouped by Gender')
sns.boxplot( merging['AudSal'], merging['GenderSelect'], data = merging)
```

Out[25]: <matplotlib.axes._subplots.AxesSubplot at 0x15341b3fe48>



30. What do you notice about the distributions?

Answer

The Median for all the three genders distribution lies almost around same value Although there is difference between the maximum salary of Male (reaching out to near about 800000) and its counterpart Female (just about 300000).

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 [130]:
```
# Your code

nations = merging['Country']
#aggregate function to calculate the median salary, utlising CompensationAmount column for salary
fun = {'AudSal':{'Medain Salary': 'median'}}
fun
#grouping them as per Gender and slicing as per the country sepecified
SalData = merging.groupby(['Country','GenderSelect']).agg(fun).loc[['Australia','India','United States'],:]
SalData = SalData.reset_index() #setting the indexing
SalData.columns = SalData.columns.droplevel(0) #dropping the column level to zero
SalData.rename(columns = {'':'Country'},inplace = True) # renaming the first column
SalData
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[130]:

	Country	Country	Medain Salary
0	Australia	Female	82000.000000
1	Australia	Male	130000.000000
2	India	A different identity	13628.148800
3	India	Female	12654.709600
4	India	Male	17327.217760
5	United States	A different identity	168264.137295
6	United States	Female	112176.091530
7	United States	Male	143336.116955

4.4 Salary and formal education

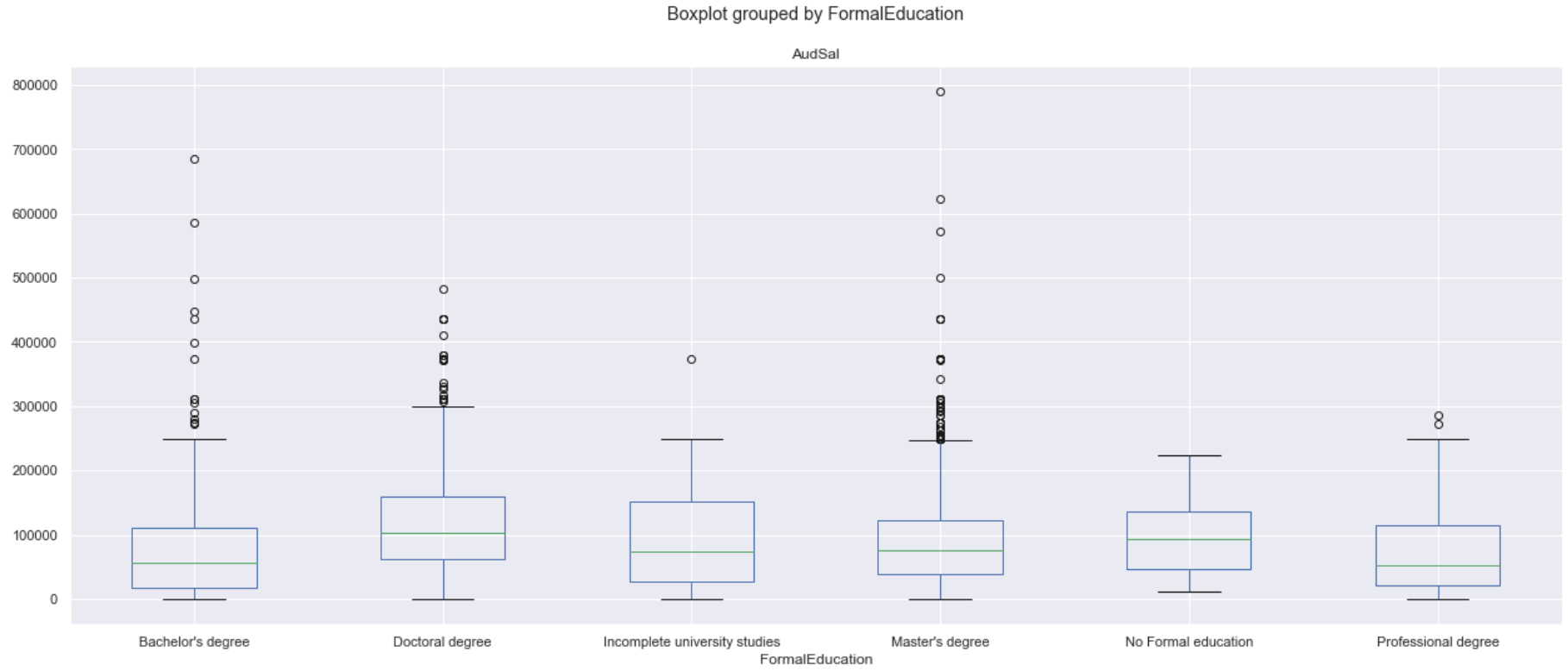
Is getting your master's really worth it ? Do PhDs get more money?

Let's see.

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

In [81]:
```
#salary distribution according to Formal Education
sal = merging['AudSal']
#boxplot groupby FormalEducation
#using the merging file, that we have obtained after merging both csv's
#and ploting the box plot
merging.boxplot(column = 'AudSal', by = 'FormalEducation')
```

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x18fdcd8a9e8>



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

Answer As per the above boxplot, it is pretty evident that the median value of salary of Master's Degree is higher than Bachelor's but Doctral Degree holder have maximum median salary. At the same time the maximum salary when compared between these three degree holders turned out to be achieved by Master's Degree holder. So, getting a MASTER'S Degree is the better Choice

4.5 Salary and job

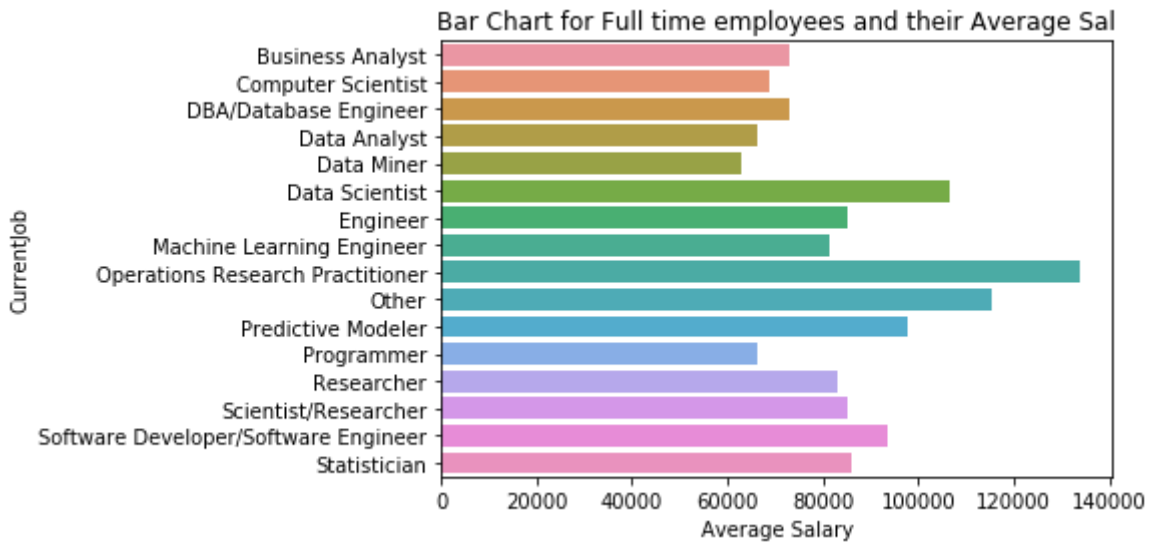
So are data scientists the highest paid in the industry? Or are there lesser known roles that are hiding from the spotlight?

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

In [100]:
```
#using the merging file, that we have obtained after merging both csv's

FullTimers = merging['EmploymentStatus'] == 'Employed full-time'
merging["new"] = FullTimers #inserting new series in response dataframe
EmployementT = merging[FullTimers] #storing result for EmployementT
#EmployementT

#calculating average salary
avg = {'AudSal': {'Average Salary': 'mean'}}
ET = EmployementT.groupby('CurrentJobTitleSelect').agg(avg).reset_index() #resetting the indexes
ET.columns = ET.columns.droplevel(0) #dropping the column level to zero
ET.rename(columns = {'':'CurrentJob'},inplace = True) # renaming the first column as CurrentJob
#plotting the barplot for the Full-time employees as per their job title
sns.barplot(ET['Average Salary'], ET['CurrentJob'], data = ET) #plotting the barplot using seaborn library
plt.title('Bar Chart for Full time employees and their Average Sal') #title for the bar plot
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[100]: Text(0.5,1,'Bar Chart for Full time employees and their Average Sal')



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

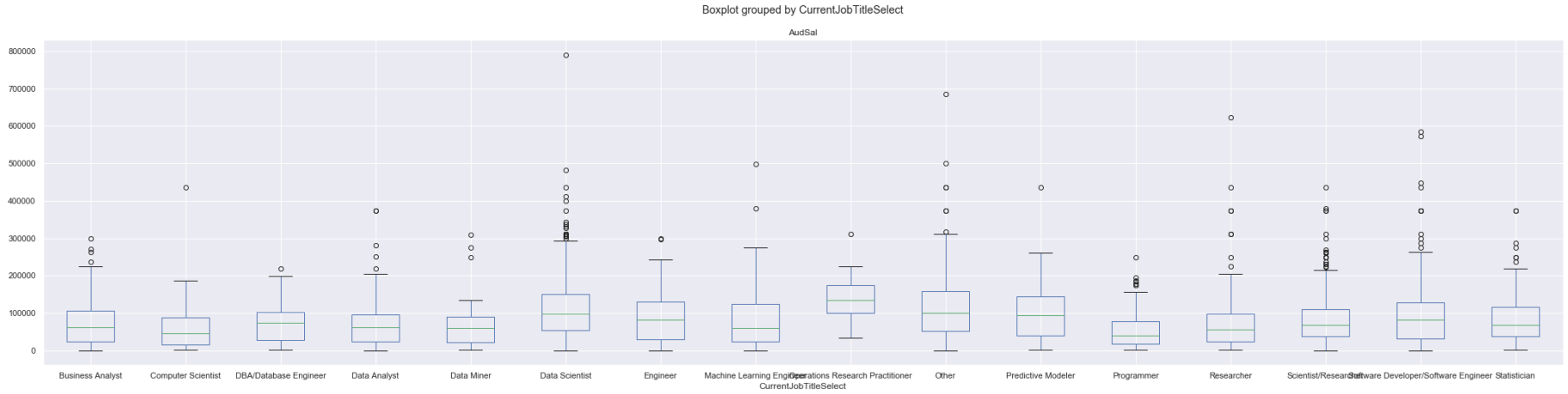
Answer

Average Salary of Engineers is highest amoung all full timers. Which is aprroximately 9.208013e+06 = 9208013, followed by computer scientists.

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 [133]:
```
##using the merging file, that we have obtained after merging both csv's
FullTimers = merging['EmploymentStatus'] == 'Employed full-time'
merging["new"] = FullTimers #inserting new series in merging dataframe
EmployementT = merging[FullTimers] #storing result for EmployementT
#boxplot for fulltimers grouped by job title
merging.boxplot(column = 'AudSal', by = 'CurrentJobTitleSelect',figsize = (35,8))
```

Out[133]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x18ff0140da0>
```



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

Answer

This could be because of the fact that Data scientists are earning more than Data analyst , database Engineer and even from any other full time Computer scientist

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 [82]:
```
fig, ax = plt.subplots()
#scatter plot with x and y axis respectively
Scatter_plot = ax.scatter( merging['AudSal'],
                          merging['Age']) #using the 'merging' dataframe, that we have obtained after merging both csv's
ax.set_xlabel('Salary Distribution') #labelling x axis
ax.set_ylabel('Age Of Respondants')  #labelling y axis
ax.set_title('Scattered Plot Salary Vs Age Of Respondants') #setting title for the scattered plot
plt.show()
```

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 [83]:
```
#creating a new dataframe that consists of emplyementype as Full-time only
FullTimers = merging['EmploymentStatus'] == 'Employed full-time'
merging["new"] = FullTimers #inserting new series in response dataframe
EmployementT = merging[FullTimers] #storing result for EmployementT

sns.set(color_codes=True)
#ploting the regression plot with x as sal in AUD and y as Age
sns.regplot(x = EmployementT['AudSal'], y = EmployementT['Age'], data = EmployementT, fit_reg=True)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\scipy\stats\stats.py:1713: FutureWarning: Using a non-tuple sequence for multidimensional indexing is deprecated; use `arr[tuple(seq)]` instead of `arr[seq]`. In the future this will be interpreted as an array index, `arr[np.array(seq)]`, which will result either in an error or a different result.
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```

Out[83]: <matplotlib.axes._subplots.AxesSubplot at 0x18fdcf9de10>

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

Answer

It can be depicted that after over 25 Years of age there is an increase in the salary of the employees

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.

In [4]:
```
import pandas as pd   # importing the library pandas and referencing it as pd
                      # pandas are software programming library used in python for data manupulationa and analysis
import seaborn as sns # importing the library seaborn and reference it as sns, used for statistical graphics in python
import matplotlib.pyplot as plt

#to view the plots in the jupyter notebook itself using matplotlib
%matplotlib inline
```

In [5]:
```
#importing the csv file
df = pd.read_csv('healthexpenditurebyareaandsource.csv', sep = ',')
df.head(5)
```

Out[5]:

	financial_year	state	area_of_expenditure	broad_source_of_funding	detailed_source_of_funding	real_expenditure_millions
0	1997-98	NSW	Administration	Government	Australian Government	315.0
1	1997-98	NSW	Administration	Government	State and local	120.0
2	1997-98	NSW	Administration	Non-government	Private health insurance funds	314.0
3	1997-98	NSW	Aids and appliances	Government	Australian Government	65.0
4	1997-98	NSW	Aids and appliances	Non-government	Individuals	168.0

In [6]:
```
#show the expenditure for each  states
#aggreation function for counting the expenses
function = {'real_expenditure_millions':{'EXPENSES':'count'}}
group = df.groupby('state').agg(function).reset_index() #grouping by state name then resetting the indexes
group.columns = group.columns.droplevel(0) #dropping the column level to zero
group.rename(columns = {'':'STATE'},inplace = True)#renaming forst column as STATES
group
```

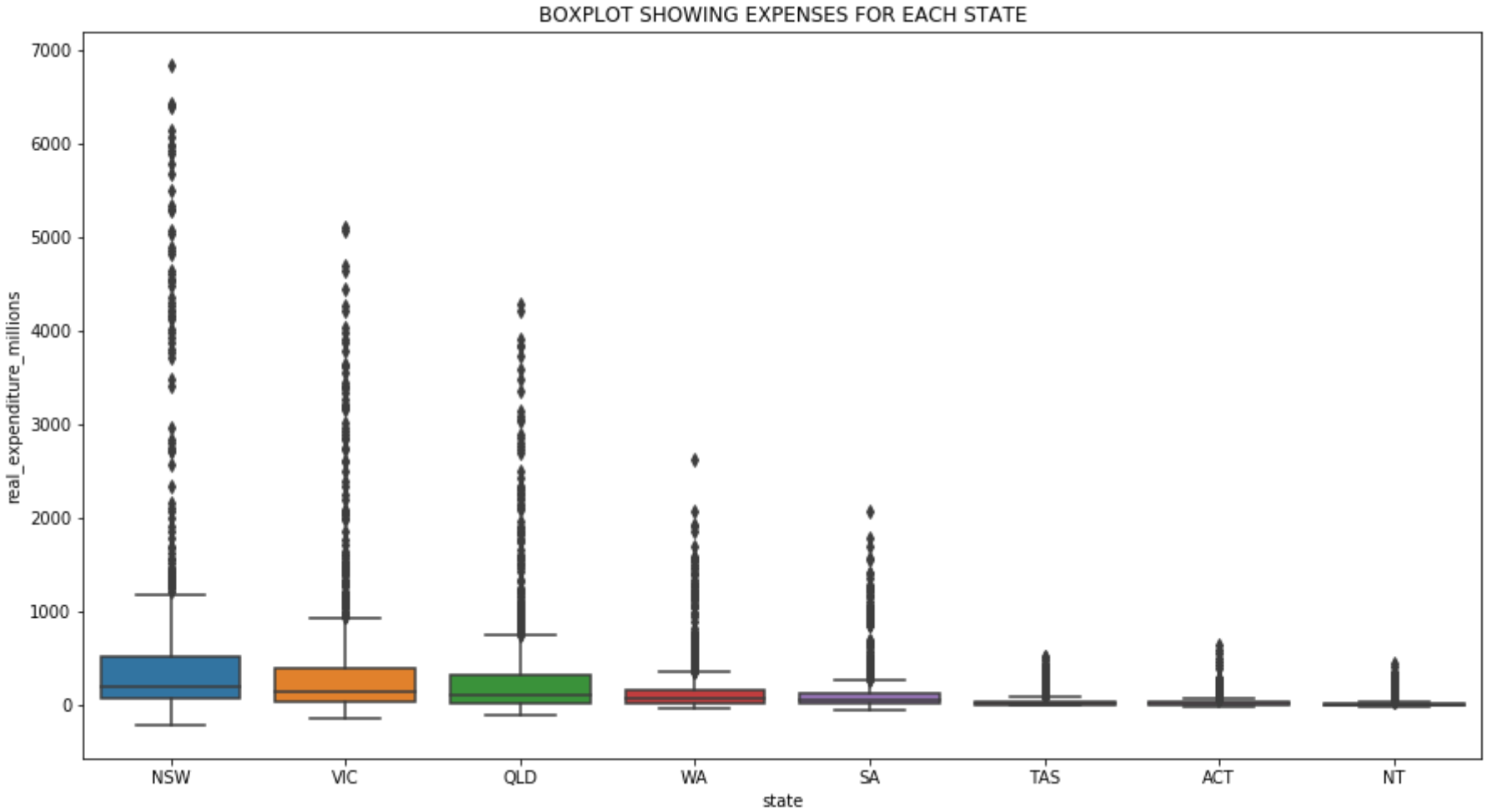
```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[6]:

	STATE	EXPENSES
0	ACT	837
1	NSW	843
2	NT	851
3	QLD	851
4	SA	861
5	TAS	838
6	VIC	833
7	WA	864

In [7]:
```
#now we have got the expenses state wise : plot the expenditures for each state
plt.figure(figsize = (15,8))
#box plot based on x and y axis
sns.boxplot(x = 'state', y = 'real_expenditure_millions', data = df)
plt.title('BOXPLOT SHOWING EXPENSES FOR EACH STATE') #ttile for the plot
```

Out[7]: Text(0.5,1,'BOXPLOT SHOWING EXPENSES FOR EACH STATE')



In [8]:
```
#Print the median expenses of victoria and QLD  grouped by detailed source of funding.

fun  = {'real_expenditure_millions':{'Expense':'median'}}
group = df.groupby(['state','detailed_source_of_funding']).agg(fun).loc[['VIC','QLD'],:]
group.reset_index()
group.columns = group.columns.droplevel(0)
group.rename(columns ={'':'State'},inplace = True)
print(group)
```

```
                                      Expense
  state detailed_source_of_funding
  QLD   Australian Government          152.0
        Individuals                    259.0
        Other non-government            22.0
        Private health insurance funds  59.5
        State and local                188.0
  VIC   Australian Government          180.0
        Individuals                    365.0
        Other non-government            67.0
        Private health insurance funds  92.5
        State and local                199.0

  C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
  ing is deprecated and will be removed in a future version
    return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

In [9]:
```
#print the average expenditure of each state on year by year basis
fun  = {'real_expenditure_millions':{'Expenses':'mean'}}
group = df.groupby(['state','financial_year']).agg(fun).reset_index()
group.columns = group.columns.droplevel(0)
#group.columns={'','year'},inplace = True)
group.columns.values[0] = 'State' #re-naming first column as state
group.columns.values[1] = 'Year' #re-naming second column as year
group
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:4656: FutureWarning: using a dict with renam
ing is deprecated and will be removed in a future version
  return super(DataFrameGroupBy, self).aggregate(arg, *args, **kwargs)
```

Out[9]:

	State	Year	Expenses
0	ACT	1997-98	26.066667
1	ACT	1998-99	24.703704
2	ACT	1999-00	28.345455
3	ACT	2000-01	25.909091
4	ACT	2001-02	27.875000
5	ACT	2002-03	30.303571
6	ACT	2003-04	30.879310
7	ACT	2004-05	32.948276
8	ACT	2005-06	33.810345
9	ACT	2006-07	34.724138
10	ACT	2007-08	35.706897
11	ACT	2008-09	39.963636
12	ACT	2009-10	41.781818
13	ACT	2010-11	43.482759
14	ACT	2011-12	47.327586
15	NSW	1997-98	426.547170
16	NSW	1998-99	426.333333

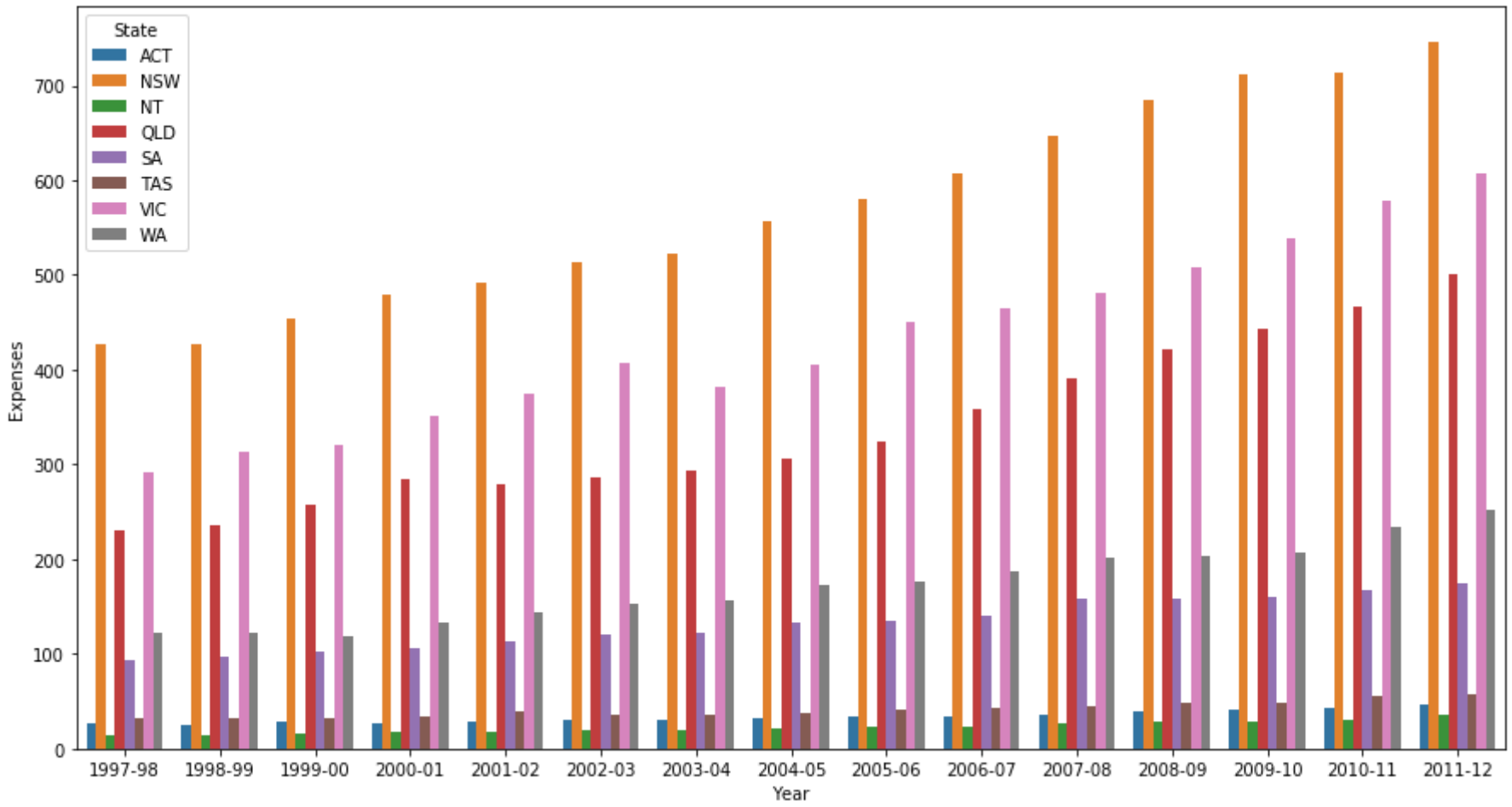
	State	Year	Expenses
17	NSW	1999-00	453.351852
18	NSW	2000-01	479.963636
19	NSW	2001-02	491.857143
20	NSW	2002-03	514.250000
21	NSW	2003-04	523.103448
22	NSW	2004-05	556.775862
23	NSW	2005-06	579.910714
24	NSW	2006-07	608.017857
25	NSW	2007-08	646.464286
26	NSW	2008-09	684.285714
27	NSW	2009-10	711.267857
28	NSW	2010-11	714.086207
29	NSW	2011-12	746.724138
...	...	...	...
90	VIC	1997-98	292.537037
91	VIC	1998-99	313.509434
92	VIC	1999-00	320.290909
93	VIC	2000-01	351.290909
94	VIC	2001-02	373.839286
95	VIC	2002-03	407.750000
96	VIC	2003-04	381.534483
97	VIC	2004-05	405.086207
98	VIC	2005-06	450.518519
99	VIC	2006-07	465.127273
100	VIC	2007-08	480.181818
101	VIC	2008-09	508.857143

	State	Year	Expenses
102	VIC	2009-10	539.107143
103	VIC	2010-11	579.160714
104	VIC	2011-12	607.767857
105	WA	1997-98	121.452830
106	WA	1998-99	122.200000
107	WA	1999-00	118.741379
108	WA	2000-01	133.309091
109	WA	2001-02	143.436364
110	WA	2002-03	153.607143
111	WA	2003-04	157.206897
112	WA	2004-05	171.948276
113	WA	2005-06	175.655172
114	WA	2006-07	186.620690
115	WA	2007-08	201.465517
116	WA	2008-09	203.655738
117	WA	2009-10	207.065574
118	WA	2010-11	234.116667
119	WA	2011-12	252.166667

120 rows × 3 columns

In [10]:
```
#bar plot of the average expenses by each state for respective financial year
plt.figure(figsize = (15,8))
#barplot for average expense
#x asis as year, y as Expenses
sns.barplot(x = 'Year', y = 'Expenses', hue = 'State', data = group)
```

Out[10]: <matplotlib.axes._subplots.AxesSubplot at 0x1f0651588d0>



In [11]:
```
#  1.From the above graph it is pretty evident that New South wales's health expenses are maximum and
#  have been growing up since 1997-98
#  2.Followed by South Australia and then Queensland
# let's ge the better idea and try to explore this graph using regression plot
#df.drop(df.tail(300).index,inplace=True) # drop last n rows
df.dropna()
df.dropna(how='all')
```

Out[11]:

	financial_year	state	area_of_expenditure	broad_source_of_funding	detailed_source_of_funding	real_expenditure_millions
0	1997-98	NSW	Administration	Government	Australian Government	315.0
1	1997-98	NSW	Administration	Government	State and local	120.0
2	1997-98	NSW	Administration	Non-government	Private health insurance funds	314.0
3	1997-98	NSW	Aids and appliances	Government	Australian Government	65.0
4	1997-98	NSW	Aids and appliances	Non-government	Individuals	168.0
5	1997-98	NSW	Aids and appliances	Non-government	Other non-government	18.0
6	1997-98	NSW	Aids and appliances	Non-government	Private health insurance funds	78.0
7	1997-98	NSW	All other medications	Government	Australian Government	5.0
8	1997-98	NSW	All other medications	Non-government	Individuals	559.0
9	1997-98	NSW	All other medications	Non-government	Other non-government	16.0
10	1997-98	NSW	All other medications	Non-government	Private health insurance funds	16.0
11	1997-98	NSW	Benefit-paid pharmaceuticals	Government	Australian Government	1068.0
12	1997-98	NSW	Benefit-paid pharmaceuticals	Non-government	Individuals	221.0
13	1997-98	NSW	Capital expenditure	Government	Australian Government	19.0
14	1997-98	NSW	Capital expenditure	Government	State and local	420.0
15	1997-98	NSW	Capital expenditure	Non-government	Other non-government	464.0
16	1997-98	NSW	Community health	Government	Australian Government	225.0
17	1997-98	NSW	Community health	Government	State and local	504.0
18	1997-98	NSW	Community health	Non-government	Private health insurance funds	1.0
19	1997-98	NSW	Dental services	Government	Australian Government	60.0

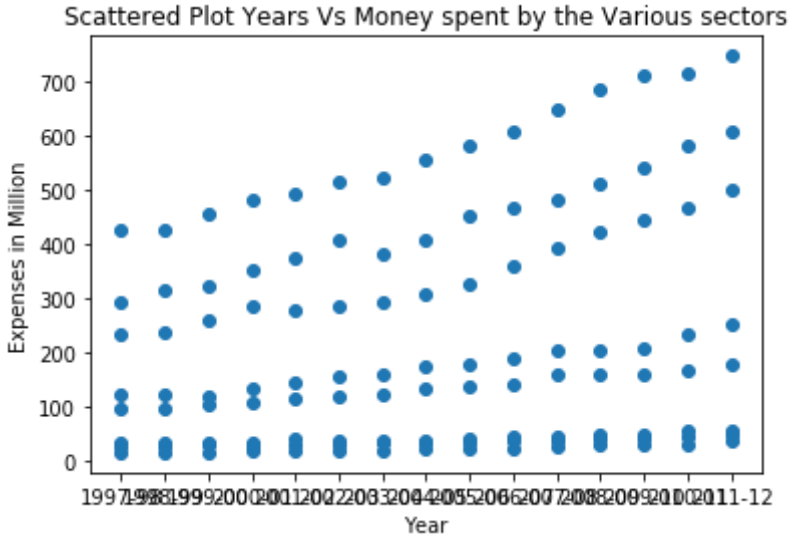
	financial_year	state	area_of_expenditure	broad_source_of_funding	detailed_source_of_funding	real_expenditure_millions
20	1997-98	NSW	Dental services	Government	State and local	156.0
21	1997-98	NSW	Dental services	Non-government	Individuals	1023.0
22	1997-98	NSW	Dental services	Non-government	Other non-government	6.0
23	1997-98	NSW	Dental services	Non-government	Private health insurance funds	351.0
24	1997-98	NSW	Medical expense tax rebate	Government	Australian Government	84.0
25	1997-98	NSW	Medical expense tax rebate	Non-government	Individuals	-84.0
26	1997-98	NSW	Medical services	Government	Australian Government	4298.0
27	1997-98	NSW	Medical services	Non-government	Individuals	522.0
28	1997-98	NSW	Medical services	Non-government	Other non-government	322.0
29	1997-98	NSW	Medical services	Non-government	Private health insurance funds	112.0
...	...	...	...	...	...	...
6748	2011-12	NT	Medical services	Non-government	Other non-government	14.0
6749	2011-12	NT	Medical services	Non-government	Private health insurance funds	5.0
6750	2011-12	NT	Other health practitioners	Government	Australian Government	8.0
6751	2011-12	NT	Other health practitioners	Government	State and local	6.0
6752	2011-12	NT	Other health practitioners	Non-government	Individuals	25.0
6753	2011-12	NT	Other health practitioners	Non-government	Other non-government	7.0
6754	2011-12	NT	Other health practitioners	Non-government	Private health insurance funds	3.0
6755	2011-12	NT	Patient transport services	Government	Australian Government	11.0
6756	2011-12	NT	Patient transport services	Government	State and local	58.0
6757	2011-12	NT	Patient transport services	Non-government	Individuals	0.0
6758	2011-12	NT	Patient transport services	Non-government	Other non-government	1.0
6759	2011-12	NT	Patient transport services	Non-government	Private health insurance funds	0.0
6760	2011-12	NT	Private hospitals	Government	Australian Government	12.0
6761	2011-12	NT	Private hospitals	Government	State and local	1.0
6762	2011-12	NT	Private hospitals	Non-government	Individuals	27.0

	financial_year	state	area_of_expenditure	broad_source_of_funding	detailed_source_of_funding	real_expenditure_millions
6763	2011-12	NT	Private hospitals	Non-government	Other non-government	6.0
6764	2011-12	NT	Private hospitals	Non-government	Private health insurance funds	24.0
6765	2011-12	NT	Public health	Government	Australian Government	35.0
6766	2011-12	NT	Public health	Government	State and local	80.0
6767	2011-12	NT	Public health	Non-government	Individuals	0.0
6768	2011-12	NT	Public health	Non-government	Other non-government	0.0
6769	2011-12	NT	Public hospitals	Government	Australian Government	202.0
6770	2011-12	NT	Public hospitals	Government	State and local	453.0
6771	2011-12	NT	Public hospitals	Non-government	Individuals	6.0
6772	2011-12	NT	Public hospitals	Non-government	Other non-government	2.0
6773	2011-12	NT	Public hospitals	Non-government	Private health insurance funds	1.0
6774	2011-12	NT	Research	Government	Australian Government	81.0
6775	2011-12	NT	Research	Government	State and local	35.0
6776	2011-12	NT	Research	Non-government	Individuals	0.0
6777	2011-12	NT	Research	Non-government	Other non-government	1.0

6778 rows × 6 columns

In [24]:
```
fig, ax = plt.subplots()
#scatter plot with x and y axis respectively
Scatter_plot = ax.scatter( group['Year'],
                          group['Expenses'])
#Scatter_plot
ax.set_xlabel('Year') #labelling x axis
ax.set_ylabel('Expenses in Million')  #labelling y axis
ax.set_title('Scattered Plot Years Vs Money spent by the Various sectors') #setting title for the scattered plot
#plt.show()
```

Out[24]: Text(0.5,1,'Scattered Plot Years Vs Money spent by the Various sectors')



1. Purpose of data: From the the above visualisation it is perceptible that the data is related to the Heath.

   The data is retrieved from:https://data.gov.au/dataset/ds-dga-f84b9baf-c1c1-437c-8c1e-654b2829848c/details?q=health (https://data.gov.au/dataset/ds-dga-f84b9baf-c1c1-437c-8c1e-654b2829848c/details?q=health)

It consists of information regdaring the expenditure of the Government of Australia in Health sector in all eight State
The data is catagories as per financial year starting from 1997 till 2012.
The funding done by the private or public sector is also listed.
The term ‘health expenditure’ in this context relates to all funds given to, or for, providers of health goods and services. It includes the funds provided by the Australian Government to the state and territory governments, as well as the funds provided by the state and territory governments to providers.

2. Ethical and Privacy Issues: The issues with the dataset could be when a person's health data is shared without his/her consent.By doing so the government is not only voilating the privacy of that indivdual but also breaking the trust.

   This may refrain patient from sharing their data accuratly.

   The Health Insurance Portability and Accountability Act (HIPAA), published in 1996, is the core set of healthcare IT data standards in the USA. The HIPAA Rules regulate the use and disclosure of personal health information (PHI) and establish national standards to protect individuals’ electronic PHI from data theft. The Health Information Technology for Economic and Clinical Health Act (HITECH Act), adopted in 2009, is aimed to “improve health care quality, safety, and efficiency through the promotion of health IT, including electronic health records and private and secure. Such approach can heal ecuring the data from theift and to gain trust of the people.

3.Environmental Benefit: It could be seen from the scattered plot that year by year the expenses are increading for health sector for almost all the sates. Which in a way also signifies that the government is spending handsome mount of money in this sector and want to make sure that the people of the country remains healthy. This can be considered as people remains healthy and hence the envirnament aroundd them aslo remain toxic free.

4.Socail Benefits: data cleaning; data formatting; the integration of different sources into a comprehensive data set; and storage using third-party tools to facilitate access and shareability could help the society and can give them easy access to the data

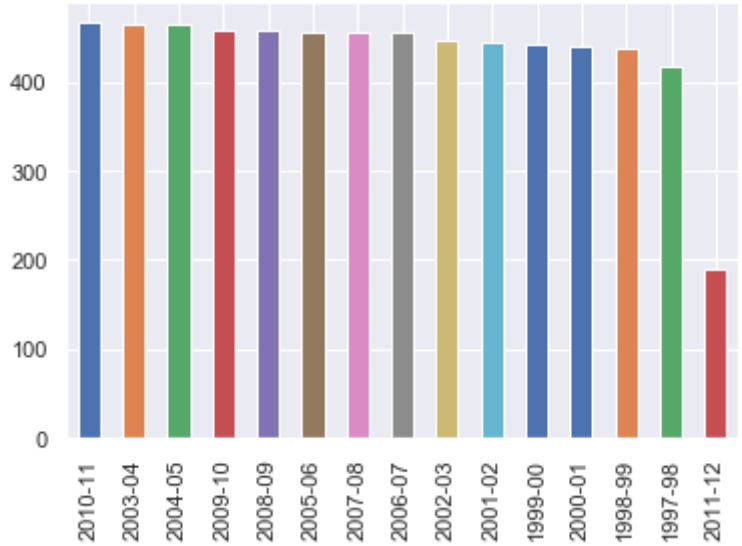
5.Health benefit: Healthcare data sets include vast amount of medical data, various measurements, financial data, statistical data, demographics of specific populations, and insurance data, to name just a few, gathered from various healthcare data sources. Due to the diversity of healthcare data sources data standardization is a key pillar for efficient and meaningful use of the information and collaboration of healthcare professionals, care providers, insurers, and government agencies. The data which i have extracted is giving the idea about the expenses and the statistics of it on yearly basis

6.Commercial Benefit: Most of the companies have the commerical data which being collected, but it is important to utilise it in proper manner the

identification of the key groups with precision play a vital role. If the data of governement of Australia Health expenditure is analysed deeply and compbined wtith some other set of data it can reveal lot of valuable information. With this in-depth knowledge, organizations can tailor services and products to customer groups, and help profit margins flourish.

In [76]:
```
pd.value_counts(df.financial_year).plot(kind = 'bar')
#response count for each financial year
```

Out[76]: <matplotlib.axes._subplots.AxesSubplot at 0x1b4ca9785c0>



In [ ]:

In [ ]:

In [ ]: