In [ ]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

```
/usr/local/lib/python3.6/dist-packages/statsmodels/tools/_testing.py:19: FutureWarning: p
andas.util.testing is deprecated. Use the functions in the public API at pandas.testing i
nstead.
  import pandas.util.testing as tm
```

In [ ]:

```
df = pd.read_csv('WA_Fn-UseC_-HR-Employee-Attrition.csv')
```

In [ ]:

```
df.head()
```

Out[ ]:

| | Age | Attrition | BusinessTravel | DailyRate | Department | DistanceFromHome | Education | EducationField | EmployeeCount |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 41 | Yes | Travel_Rarely | 1102 | Sales | 1 | 2 | Life Sciences | 1 |
| 1 | 49 | No | Travel_Frequently | 279 | Research & Development | 8 | 1 | Life Sciences | 1 |
| 2 | 37 | Yes | Travel_Rarely | 1373 | Research & Development | 2 | 2 | Other | 1 |
| 3 | 33 | No | Travel_Frequently | 1392 | Research & Development | 3 | 4 | Life Sciences | 1 |
| 4 | 27 | No | Travel_Rarely | 591 | Research & Development | 2 | 1 | Medical | 1 |

In [ ]:

```
df.shape
```

Out[ ]:

```
(1470, 35)
```

# HR Attrition dataset

**The HR Attrition dataset contains the Attrition rate based on various parameters / Variables in a company.**

1. **This dataset contains 35 variable with 1470 rows.**
2. **In this dataset, the Attrition is the Dependent varible and rest 34 columns are the Independent variable.**

**The analysis of Independent variables with respect to Dependent variable is done and the dataset is next prepared for ML to predict Attrition rate.**

In [ ]:

```
# Description of the data
df.describe()
```

Out[ ]:

| | Age | DailyRate | DistanceFromHome | Education | EmployeeCount | EmployeeNumber | EnvironmentSatisfactio |
|---|---|---|---|---|---|---|---|

| | Age | DailyRate | DistanceFromHome | Education | EmployeeCount | EmployeeNumber | EnvironmentSatisfactio |
|---|---|---|---|---|---|---|---|
| count | 1470.000000 | 1470.000000 | 1470.000000 | 1470.000000 | 1470.0 | 1470.000000 | 1470.00000 |
| mean | 36.923810 | 802.485714 | 9.192517 | 2.912925 | 1.0 | 1024.865306 | 2.72176 |
| std | 9.135373 | 403.509100 | 8.106864 | 1.024165 | 0.0 | 602.024335 | 1.09308 |
| min | 18.000000 | 102.000000 | 1.000000 | 1.000000 | 1.0 | 1.000000 | 1.00000 |
| 25% | 30.000000 | 465.000000 | 2.000000 | 2.000000 | 1.0 | 491.250000 | 2.00000 |
| 50% | 36.000000 | 802.000000 | 7.000000 | 3.000000 | 1.0 | 1020.500000 | 3.00000 |
| 75% | 43.000000 | 1157.000000 | 14.000000 | 4.000000 | 1.0 | 1555.750000 | 4.00000 |
| max | 60.000000 | 1499.000000 | 29.000000 | 5.000000 | 1.0 | 2068.000000 | 4.00000 |

**From the dataset description, we can observe following things.**

1. **The count for all the variables are same and hence we can say that there are no null values present although we shall check for null values separately for confirmation.**
2. **The min age and max age for work is 18 and 60 which gives the legal age for work and retirement. So, the age column is said to be correct.**
3. **The minimum and maximum monthly income is around 1000 and 20000 units.**
4. **The minimum and max hourly rate is around 30 and 100.**

In [ ]:

```
# Verification of null values
df.isnull().any()
```

Out[ ]:

```
Age                         False
Attrition                   False
BusinessTravel              False
DailyRate                   False
Department                  False
DistanceFromHome            False
Education                   False
EducationField              False
EmployeeCount               False
EmployeeNumber              False
EnvironmentSatisfaction     False
Gender                      False
HourlyRate                  False
JobInvolvement              False
JobLevel                    False
JobRole                     False
JobSatisfaction             False
MaritalStatus               False
MonthlyIncome               False
MonthlyRate                 False
NumCompaniesWorked          False
Over18                      False
OverTime                    False
PercentSalaryHike           False
PerformanceRating           False
RelationshipSatisfaction    False
StandardHours               False
StockOptionLevel            False
TotalWorkingYears           False
TrainingTimesLastYear       False
WorkLifeBalance             False
YearsAtCompany              False
YearsInCurrentRole          False
YearsSinceLastPromotion     False
YearsWithCurrManager        False
dtype: bool
```

**This gives the confirmation that there are no null values for overall dataset.**

In [ ]:

```
# Datatyes for each attributes/variables
df.dtypes
```

Out[ ]:

```
Age                          int64
Attrition                   object
BusinessTravel              object
DailyRate                    int64
Department                  object
DistanceFromHome             int64
Education                    int64
EducationField              object
EmployeeCount                int64
EmployeeNumber               int64
EnvironmentSatisfaction      int64
Gender                      object
HourlyRate                   int64
JobInvolvement               int64
JobLevel                     int64
JobRole                     object
JobSatisfaction              int64
MaritalStatus               object
MonthlyIncome                int64
MonthlyRate                  int64
NumCompaniesWorked           int64
Over18                      object
OverTime                    object
PercentSalaryHike            int64
PerformanceRating            int64
RelationshipSatisfaction     int64
StandardHours                int64
StockOptionLevel             int64
TotalWorkingYears            int64
TrainingTimesLastYear        int64
WorkLifeBalance              int64
YearsAtCompany               int64
YearsInCurrentRole           int64
YearsSinceLastPromotion      int64
YearsWithCurrManager         int64
dtype: object
```

**The data types for each varaible is presented and by observation we can say that these data types are said to be correct and no conversion needed.**

In [ ]:

```
# Attrition counts (Employees currently in company)
print(df['Attrition'].value_counts())
print(df['Attrition'].value_counts() / df['Attrition'].value_counts().sum()*100)
```

```
No     1233
Yes     237
Name: Attrition, dtype: int64
No     83.877551
Yes    16.122449
Name: Attrition, dtype: float64
```

**From the value counts of Attrition rate, we can observe that there are around 84% of employees who have left and company and 16% remaining.**

In [ ]:

```
# Attrition rate based on age
df['Age'].unique()
```

Out[ ]:

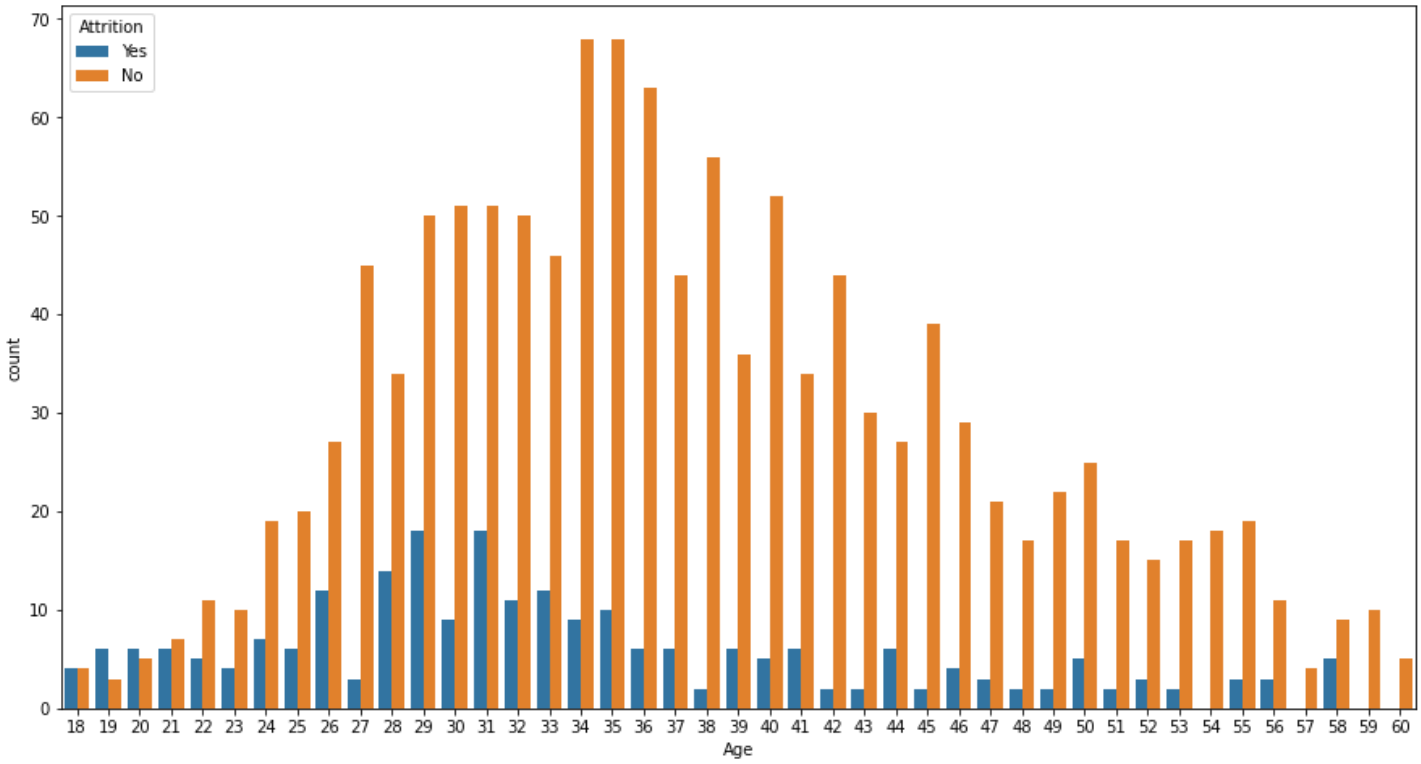
```
array([41, 49, 37, 33, 27, 32, 59, 30, 38, 36, 35, 29, 31, 34, 28, 22, 53,
       24, 21, 42, 44, 46, 39, 43, 50, 26, 48, 55, 45, 56, 23, 51, 40, 54,
       58, 20, 25, 19, 57, 52, 47, 18, 60])
```

In [ ]:

```
#df.groupby('Age')['Attrition'].values
```

In [ ]:

```
#sns.countplot(x = 'Age', hue = "smoker", data = df)
#plt.plot(figsize = (10, 15))
plt.subplots(figsize = (15, 8))
sns.countplot(x = 'Age', hue = "Attrition", data = df)
plt.show()
```




**From sns.countplot, we can address two variables in a single bar graph. The above graph indicates the Attrition rate for Age variable. The following observations can be made.**

1. **The attrition rate "No" (Employees not present in company) is highest for age group of 34 and 35 followed by 36.**
2. **For age 29 and 31, the attriition rate "Yes" shows highest.**
3. **Although the attrition rate "Yes" is high for few Age group, apart from age 18-20, the attrition rate "No" is dominating the attrition rate "Yes".**

In [ ]:

```
df.head()
```

Out[ ]:

| | Age | Attrition | BusinessTravel | DailyRate | Department | DistanceFromHome | Education | EducationField | EmployeeCount |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 41 | Yes | Travel_Rarely | 1102 | Sales | 1 | 2 | Life Sciences | 1 |
| 1 | 49 | No | Travel_Frequently | 279 | Research & Development | 8 | 1 | Life Sciences | 1 |
| 2 | 37 | Yes | Travel_Rarely | 1373 | Research & Development | 2 | 2 | Other | 1 |
| 3 | 33 | No | Travel_Frequently | 1392 | Research & Development | 3 | 4 | Life Sciences | 1 |

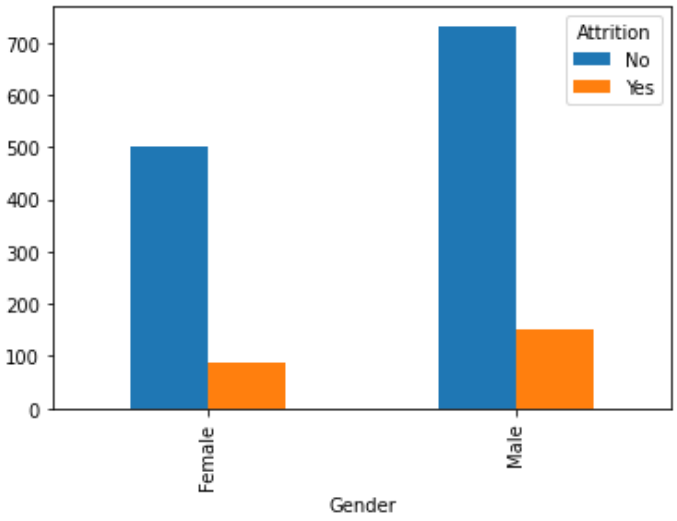
| | Age | Attrition | BusinessTravel | DailyRate | Department | DistanceFromHome | Education | EducationField | EmployeeCount |
|---|---|---|---|---|---|---|---|---|---|
| 4 | 27 | No | Travel_Rarely | 591 | Research & Development | 2 | 1 | Medical | 1 |

In [ ]:

```
data = df.groupby(['Gender', 'Attrition'])['Attrition'].count()
data.unstack().plot.bar()

# Percentage calculation
d_f = (df[df['Gender'] == 'Female'].count()[0] / df.shape[0])*100
d_m = (df[df['Gender'] == 'Male'].count()[0] / df.shape[0])*100
print(data)
print('Female with No Attrition: ', data.values[0] / data.values[0:2].sum()*100, '%')
print('Female with Yes Attrition: ', data.values[1] / data.values[0:2].sum()*100, '%')
print('Male with No Attrition: ', data.values[2] / data.values[2:4].sum()*100, '%')
print('Male with Yes Attrition: ', data.values[3] / data.values[2:4].sum()*100, '%')
print('Female Percentage : ',d_f,'%')
print('Male Percentage : ',d_m,'%')
"""
gender = data.values
labels = 'Female', 'Male'
plt.pie(gender, labels = labels, autopct = '%0.f%%')
"""
plt.show()
```

```
Gender  Attrition
Female  No           501
        Yes           87
Male    No           732
        Yes          150
Name: Attrition, dtype: int64
Female with No Attrition:  85.20408163265306 %
Female with Yes Attrition:  14.795918367346939 %
Male with No Attrition:  82.99319727891157 %
Male with Yes Attrition:  17.006802721088434 %
Female Percentage :  40.0 %
Male Percentage :  60.0 %
```

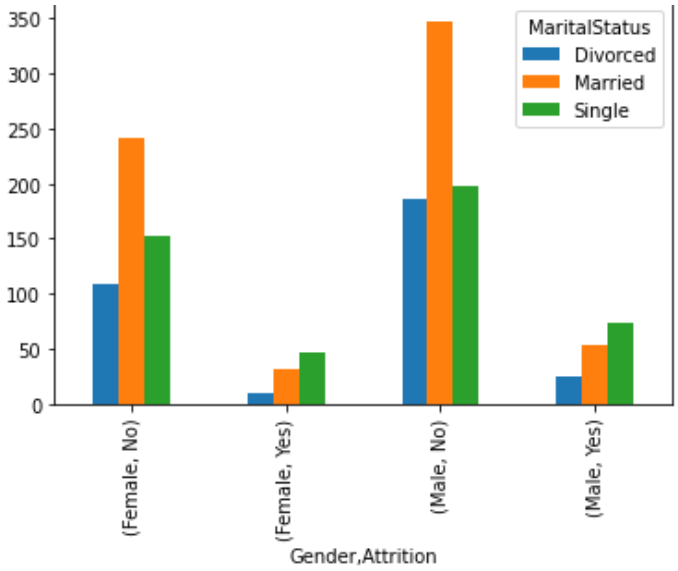



**From the above analysis of Gender and Attrition, we can say that**

1. **The company have 40% female employees and 60% Male employees.**
2. **From bar graph, we can observe that out of 40% female strength, 85% Female has Attrition "No" and rest 15% has Attrition "Yes".**
3. **Similarly, out of 60% Male strength, 83% Male has Attrition "No" and rest 17% has Attrition "Yes".**

In [ ]:

```
data = df.groupby(['Gender', 'Attrition', 'MaritalStatus'])['Attrition'].count()
data.unstack().plot.bar()
plt.show()
```




**From the graph, we can observe that:**

1. **The Attrition rate "No" for both Male and Female Employees is higher for Married status following with Single and Divorced.**
2. **While, the Attrition rate "Yes" for both Male and Female Employees is higher for Single statur following with Married and Divorced Status.**

In [ ]:

```
for col in df.columns:
  print(str(col) + str(df[col].unique()))
```

```
Age[41 49 37 33 27 32 59 30 38 36 35 29 31 34 28 22 53 24 21 42 44 46 39 43
 50 26 48 55 45 56 23 51 40 54 58 20 25 19 57 52 47 18 60]
Attrition['Yes' 'No']
BusinessTravel['Travel_Rarely' 'Travel_Frequently' 'Non-Travel']
DailyRate[1102  279 1373 1392  591 1005 1324 1358  216 1299  809  153  670 1346
  103 1389  334 1123 1219  371  673 1218  419  391  699 1282 1125  691
  477  705  924 1459  125  895  813 1273  869  890  852 1141  464 1240
 1357  994  721 1360 1065  408 1211 1229  626 1434 1488 1097 1443  515
  853 1142  655 1115  427  653  989 1435 1223  836 1195 1339  664  318
 1225 1328 1082  548  132  746  776  193  397  945 1214  111  573 1153
 1400  541  432  288  669  530  632 1334  638 1093 1217 1353  120  682
  489  807  827  871  665 1040 1420  240 1280  534 1456  658  142 1127
 1031 1189 1354 1467  922  394 1312  750  441  684  249  841  147  528
  594  470  957  542  802 1355 1150 1329  959 1033 1316  364  438  689
  201 1427  857  933 1181 1395  662 1436  194  967 1496 1169 1145  630
  303 1256  440 1450 1452  465  702 1157  602 1480 1268  713  134  526
 1380  140  629 1356  328 1084  931  692 1069  313  894  556 1344  290
  138  926 1261  472 1002  878  905 1180  121 1136  635 1151  644 1045
  829 1242 1469  896  992 1052 1147 1396  663  119  979  319 1413  944
 1323  532  818  854 1034  771 1401 1431  976 1411 1300  252 1327  832
 1017 1199  504  505  916 1247  685  269 1416  833  307 1311  128  488
  529 1210 1463  675 1385 1403  452  666 1158  228  996  728 1315  322
 1479  797 1070  442  496 1372  920  688 1449 1117  636  506  444  950
  889  555  230 1232  566 1302  812 1476  218 1132 1105  906  849  390
  106 1249  192  553  117  185 1091  723 1220  588 1377 1018 1275  798
  672 1162  508 1482  559  210  928 1001  549 1124  738  570 1130 1192
  343  144 1296 1309  483  810  544 1062 1319  641 1332  756  845  593
 1171  350  921 1144  143 1046  575  156 1283  755  304 1178  329 1362
 1371  202  253  164 1107  759 1305  982  821 1381  480 1473  891 1063
  645 1490  317  422 1485 1368 1448  296 1398 1349  986 1099 1116 1499
  983 1009 1303 1274 1277  587  413 1276  988 1474  163  267  619  302
  443  828  561  426  232 1306 1094  509  775  195  258  471  799  956
  535 1495  446 1245  703  823 1246  622 1287  448  254 1365  538  525
  558  782  362 1236 1112  204 1343  604 1216  646  160  238 1397  306
  991  482 1176  913 1076  727  885  243  806  817 1410 1207 1442  693
  929  562  608  580  970 1179  294  314  316  654  168  381  217  501
  650  141  804  975 1090  346  430  268  167  621  527  883  954  310
  719  725  715  657 1146  182  376  571  384  791 1111 1243 1092 1325
  805  213  118  676 1252  286 1258  932 1041  859  720  946 1184  436
```

```
  589  760  887 1318  625  180  586 1012  661  930  342 1230 1271 1278
  607  130  300  583 1418 1269  379  395 1265 1222  341  868 1231  102
  881 1383 1075  374 1086  781  177  500 1425 1454  617 1085  995 1122
  618  546  462 1198 1272  154 1137 1188  188 1333  867  263  938  129
  616  498 1404 1053  289 1376  231  152  882  903 1379  335  722  461
  974 1126  840 1134  248  955  939 1391 1206  287 1441  109 1066  277
  466 1055  265  135  247 1035  266  145 1038 1234 1109 1089  788  124
  660 1186 1464  796  415  769 1003 1366  330 1492 1204  309 1330  469
  697 1262 1050  770  406  203 1308  984  439  793 1451 1182  174  490
  718  433  773  603  874  367  199  481  647 1384  902  819  862 1457
  977  942 1402 1421 1361  917  200  150  179  696  116  363  107 1465
  458 1212 1103  966 1010  326 1098  969 1167  694 1320  536  373  599
  251  131  237 1429  648  735  531  429  968  879  640  412  848  360
 1138  325 1322  299 1030  634  524  256 1060  935  495  282  206  943
  523  507  601  855 1291 1405 1369  999 1202  285  404  736 1498 1200
 1439  499  205  683 1462  949  652  332 1475  337  971 1174  667  560
  172  383 1255  359  401  377  592 1445 1221  866  981  447 1326  748
  990  405  115  790  830 1193 1423  467  271  410 1083  516  224  136
 1029  333 1440  674 1342  898  824  492  598  740  888 1288  104 1108
  479 1351  474  437  884 1370  264 1059  563  457 1313  241 1015  336
 1387  170  208  671  711  737 1470  365  763  567  486  772  301  311
  584  880  392  148  708 1259  786  370  678  146  581  918 1238  585
  741  552  369  717  543  964  792  611  176  897  600 1054  428  181
  211 1079  590  305  953  478 1375  244  511 1294  196  734 1239 1253
 1128 1336  234  766  261 1194  431  572 1422 1297  574  355  207  706
  280  726  414  352 1224  459 1254 1131  835 1172 1266  783  219 1213
 1096 1251 1394  605 1064 1337  937  157  754 1168  155 1444  189  911
 1321 1154  557  642  801  161 1382 1037  105  582  704  345 1120 1378
  468  613 1023  628]
Department['Sales' 'Research & Development' 'Human Resources']
DistanceFromHome[ 1  8  2  3 24 23 27 16 15 26 19 21  5 11  9  7  6 10  4 25 12 18 29 22
 14 20 28 17 13]
Education[2 1 4 3 5]
EducationField['Life Sciences' 'Other' 'Medical' 'Marketing' 'Technical Degree'
 'Human Resources']
EmployeeCount[1]
EmployeeNumber[   1    2    4 ... 2064 2065 2068]
EnvironmentSatisfaction[2 3 4 1]
Gender['Female' 'Male']
HourlyRate[ 94  61  92  56  40  79  81  67  44  84  49  31  93  50  51  80  96  78
  45  82  53  83  58  72  48  42  41  86  97  75  33  37  73  98  36  47
  71  30  43  99  59  95  57  76  87  66  55  32  52  70  62  64  63  60
 100  46  39  77  35  91  54  34  90  65  88  85  89  68  69  74  38]
JobInvolvement[3 2 4 1]
JobLevel[2 1 3 4 5]
JobRole['Sales Executive' 'Research Scientist' 'Laboratory Technician'
 'Manufacturing Director' 'Healthcare Representative' 'Manager'
 'Sales Representative' 'Research Director' 'Human Resources']
JobSatisfaction[4 2 3 1]
MaritalStatus['Single' 'Married' 'Divorced']
MonthlyIncome[5993 5130 2090 ... 9991 5390 4404]
MonthlyRate[19479 24907  2396 ...  5174 13243 10228]
NumCompaniesWorked[8 1 6 9 0 4 5 2 7 3]
Over18['Y']
OverTime['Yes' 'No']
PercentSalaryHike[11 23 15 12 13 20 22 21 17 14 16 18 19 24 25]
PerformanceRating[3 4]
RelationshipSatisfaction[1 4 2 3]
StandardHours[80]
StockOptionLevel[0 1 3 2]
TotalWorkingYears[ 8 10  7  6 12  1 17  5  3 31 13  0 26 24 22  9 19  2 23 14 15  4 29 28
 21 25 20 11 16 37 38 30 40 18 36 34 32 33 35 27]
TrainingTimesLastYear[0 3 2 5 1 4 6]
WorkLifeBalance[1 3 2 4]
YearsAtCompany[ 6 10  0  8  2  7  1  9  5  4 25  3 12 14 22 15 27 21 17 11 13 37 16 20
 40 24 33 19 36 18 29 31 32 34 26 30 23]
YearsInCurrentRole[ 4  7  0  2  5  9  8  3  6 13  1 15 14 16 11 10 12 18 17]
YearsSinceLastPromotion[ 0  1  3  2  7  4  8  6  5 15  9 13 12 10 11 14]
YearsWithCurrManager[ 5  7  0  2  6  8  3 11 17  1  4 12  9 10 15 13 16 14]
```

**From the unique value analysis, we can observe that the variables "EmployeeCount, Over18, StandartHours" has**

**only 1 unique value and it does not have much influence in the dataset. So, these variables can be removed. Also, Employee Number does not have effect for the Attrition rate.**

In [ ]:

```
df.drop(['EmployeeCount', 'Over18', 'StandardHours', 'EmployeeNumber'], axis = 1, inplace=True)
```

In [ ]:

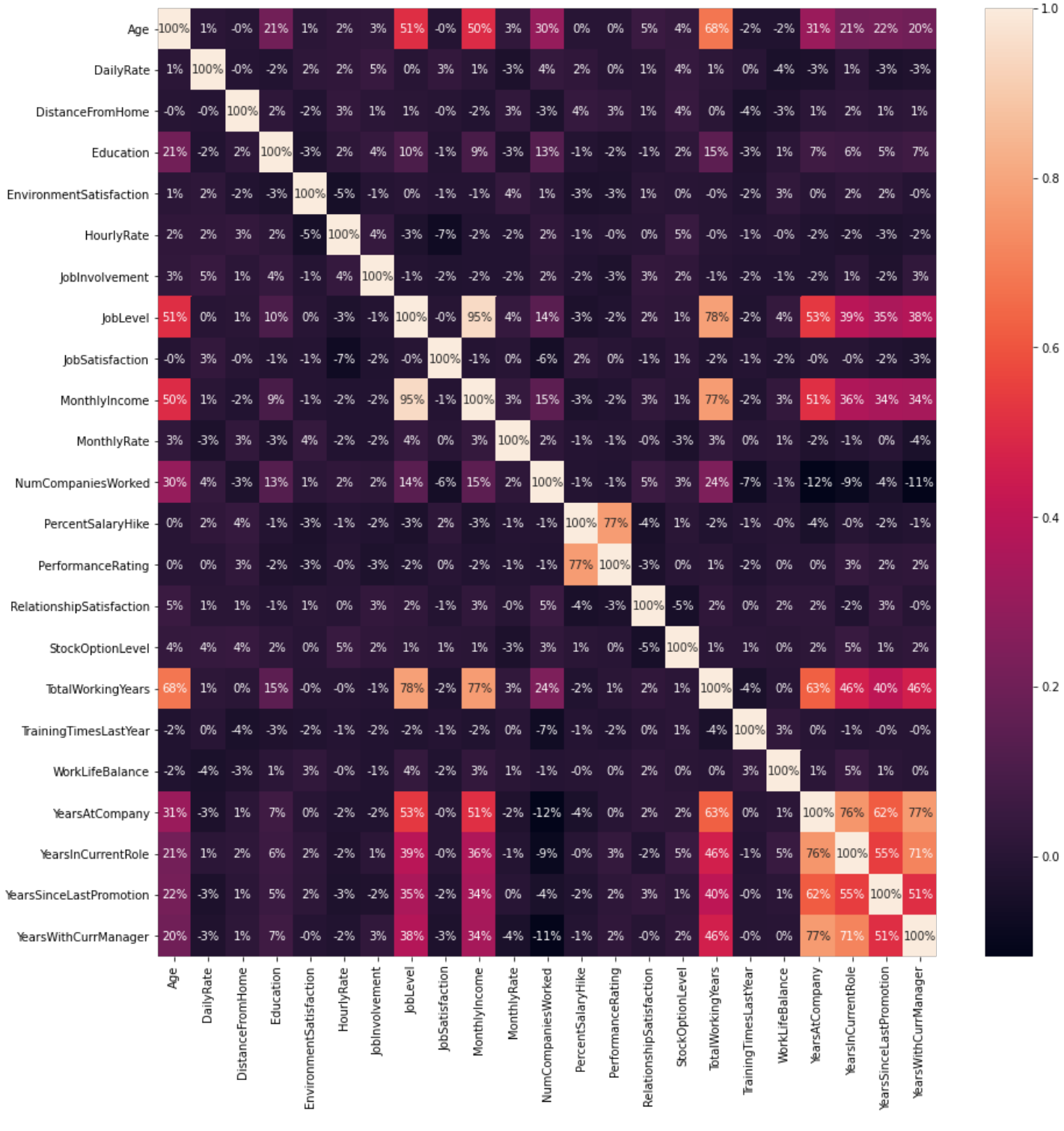
```
df.shape
```

Out[ ]:

```
(1470, 31)
```

In [ ]:

```
# Co-relation matrix
plt.figure(figsize = (15, 15))
sns.heatmap(df.corr(), annot = True, fmt = '.0%')
plt.show()
```



**From the Dataset Correlation matrix, we can observe following things:**

1. **The correlation between Job level and YearsAtCompany is at 53% which means as the working years at company increases, there is a chance of 53% increase in Job level.**
2. **As the Job level increase, there is a chance of 95% increase in Monthly income for the employees.**
3. **We can observe that there is not much negative corelation between the variables.**

# Conclusion

1. **The dataset predicts the Attrition rate of the employees.**
2. **Determination of Attrition rate based on various variables have been done. The age variable have been converted to different age groups and analysed.**
3. **Analysis have been done using correlation matrix also.**
4. **Object datatypes have to be converted to Categorical variables for the dataset to be ready for ML.**