In [1]:

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

In [3]:

```
data = pd.read_csv("resume_data.csv")
```

In [4]:

```
data.head()
```

Out[4]:

	Category	Resume
0	Data Science	Skills * Programming Languages: Python (pandas...
1	Data Science	Education Details \r\nMay 2013 to May 2017 B.E...
2	Data Science	Areas of Interest Deep Learning, Control Syste...
3	Data Science	Skills â¢ R â¢ Python â¢ SAP HANA â¢ Table...
4	Data Science	Education Details \r\n MCA YMCAUST, Faridab...

In [5]:

```
data.tail()
```

Out[5]:

	Category	Resume
957	Testing	Computer Skills: â¢ Proficient in MS office (...
958	Testing	â- Willingness to accept the challenges. â- ...
959	Testing	PERSONAL SKILLS â¢ Quick learner, â¢ Eagerne...
960	Testing	COMPUTER SKILLS & SOFTWARE KNOWLEDGE MS-Power ...
961	Testing	Skill Set OS Windows XP/7/8/8.1/10 Database MY...

In [6]:

```
data.shape
```

Out[6]:

```
(962, 2)
```

In [7]:

```
data.columns
```

Out[7]:

```
Index(['Category', 'Resume'], dtype='object')
```

In [8]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 962 entries, 0 to 961
Data columns (total 2 columns):
 #   Column    Non-Null Count  Dtype 
---  ------    --------------  ----- 
 0   Category  962 non-null    object
 1   Resume    962 non-null    object
dtypes: object(2)
memory usage: 15.2+ KB
```

In [9]:

```
data.describe()
```

Out[9]:

	Category	Resume
count	962	962
unique	25	166
top	Java Developer	Technical Skills Web Technologies: Angular JS,...
freq	84	18

In [10]:

```
data.isnull().sum()
```

Out[10]:

```
Category    0
Resume      0
dtype: int64
```

In [11]:

```
data.nunique()
```

Out[11]:

```
Category     25
Resume      166
dtype: int64
```

In [12]:

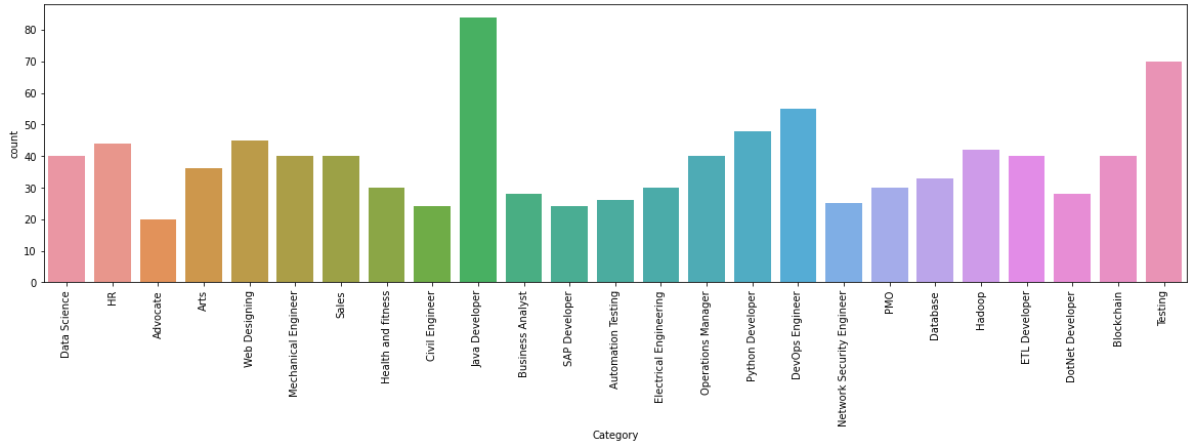
```
data['Category'].unique()
```

Out[12]:

```
array(['Data Science', 'HR', 'Advocate', 'Arts', 'Web Designing',
       'Mechanical Engineer', 'Sales', 'Health and fitness',
       'Civil Engineer', 'Java Developer', 'Business Analyst',
       'SAP Developer', 'Automation Testing', 'Electrical Engineering',
       'Operations Manager', 'Python Developer', 'DevOps Engineer',
       'Network Security Engineer', 'PMO', 'Database', 'Hadoop',
       'ETL Developer', 'DotNet Developer', 'Blockchain', 'Testing'],
      dtype=object)
```

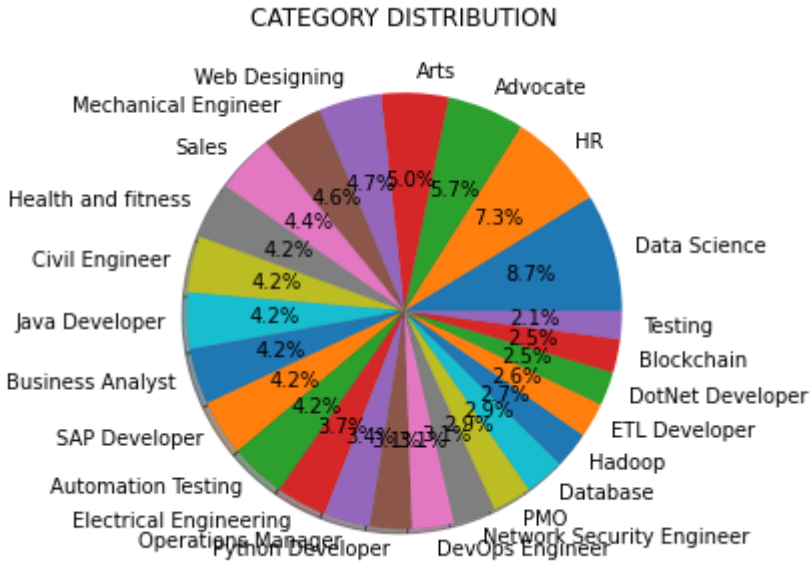
In [13]:

```
plt.figure(figsize=(20,5))
plt.xticks(rotation=90)
ax=sns.countplot(x="Category", data=data)
plt.show()
```



In [15]:

```
targetCounts = data['Category'].value_counts()
targetLabels  = data['Category'].unique()
plt.figure(figsize=(20,5))
plt.pie(targetCounts, labels=targetLabels, autopct='%1.1f%%', shadow=True)
plt.title("CATEGORY DISTRIBUTION")
plt.show()
```



In [19]:

```
import re
def cleanResume(resumeText):
    resumeText = re.sub('http\S+\s*', ' ', resumeText)  # remove URLs
    resumeText = re.sub('RT|cc', ' ', resumeText)  # remove RT and cc
    resumeText = re.sub('#\S+', '', resumeText)  # remove hashtags
    resumeText = re.sub('@\S+', '  ', resumeText)  # remove mentions
    resumeText = re.sub('[%s]' % re.escape("""!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~"""), ' ',
    resumeText = re.sub(r'[^\x00-\x7f]',r' ', resumeText)
    resumeText = re.sub('\s+', ' ', resumeText)  # remove extra whitespace
    return resumeText

data['cleaned_resume'] = data.Resume.apply(lambda x: cleanResume(x))
```

In [20]:

```
data.head()
```

Out[20]:

	Category	Resume	cleaned_resume
0	Data Science	Skills * Programming Languages: Python (pandas...	Skills Programming Languages Python pandas num...
1	Data Science	Education Details \r\nMay 2013 to May 2017 B.E...	Education Details May 2013 to May 2017 B E UIT...
2	Data Science	Areas of Interest Deep Learning, Control Syste...	Areas of Interest Deep Learning Control System...
3	Data Science	Skills â¢ R â¢ Python â¢ SAP HANA â¢ Table...	Skills R Python SAP HANA Tableau SAP HANA SQL ...
4	Data Science	Education Details \r\n MCA YMCAUST, Faridab...	Education Details MCA YMCAUST Faridabad Haryan...

In [21]:

```
data.tail()
```

Out[21]:

	Category	Resume	cleaned_resume
957	Testing	Computer Skills: â¢ Proficient in MS office (...	Computer Skills Proficient in MS office Word B...
958	Testing	â- Willingness to accept the challenges. â- ...	Willingness to a ept the challenges Positive ...
959	Testing	PERSONAL SKILLS â¢ Quick learner, â¢ Eagerne...	PERSONAL SKILLS Quick learner Eagerness to lea...
960	Testing	COMPUTER SKILLS & SOFTWARE KNOWLEDGE MS-Power ...	COMPUTER SKILLS SOFTWARE KNOWLEDGE MS Power Po...
961	Testing	Skill Set OS Windows XP/7/8/8.1/10 Database MY...	Skill Set OS Windows XP 7 8 8 1 10 Database MY...

In [22]:

```
import nltk
from nltk.corpus import stopwords
import string
from wordcloud import WordCloud
```

In [24]:

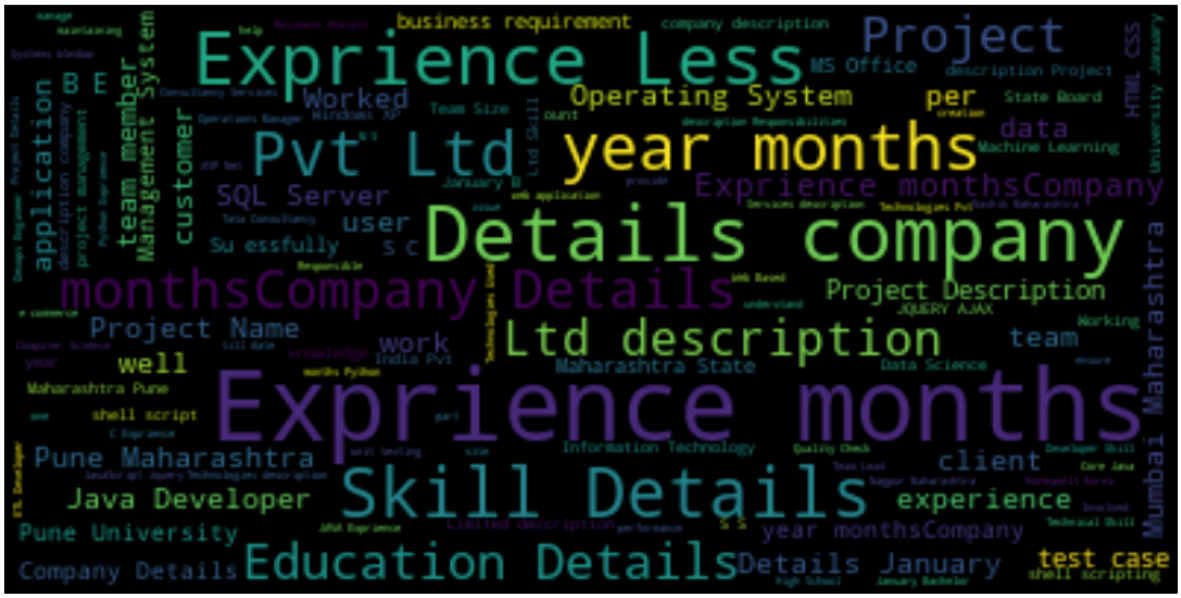
```
SetOfStopWords = set(stopwords.words('english')+['``',"''"])
totalWords =[]
Sentences = data['Resume'].values
cleanedSentences = ""
for records in Sentences:
    cleanedText = cleanResume(records)
    cleanedSentences += cleanedText
    requiredWords = nltk.word_tokenize(cleanedText)
    for word in requiredWords:
        if word not in SetOfStopWords and word not in string.punctuation:
            totalWords.append(word)

wordfreqdist = nltk.FreqDist(totalWords)
mostcommon = wordfreqdist.most_common(50)
print(mostcommon)
```

```
[('Exprience', 3829), ('months', 3233), ('company', 3130), ('Details', 296
7), ('description', 2634), ('1', 2134), ('Project', 1808), ('project', 157
9), ('6', 1499), ('data', 1438), ('team', 1424), ('Maharashtra', 1385),
('year', 1244), ('Less', 1137), ('January', 1086), ('using', 1041), ('Skil
l', 1018), ('Pune', 1016), ('Management', 1010), ('SQL', 990), ('Ltd', 93
4), ('management', 927), ('C', 896), ('Engineering', 855), ('Education', 8
33), ('Developer', 806), ('Java', 773), ('2', 754), ('development', 752),
('monthsCompany', 746), ('Pvt', 730), ('application', 727), ('System', 71
5), ('reports', 697), ('business', 696), ('India', 693), ('requirements',
693), ('I', 690), ('various', 688), ('A', 688), ('Data', 674), ('The', 67
2), ('University', 656), ('process', 648), ('Testing', 646), ('test', 63
8), ('Responsibilities', 637), ('system', 636), ('testing', 634), ('Softwa
re', 632)]
```

In [25]:

```
word_cloud = WordCloud().generate(cleanedSentences)
plt.figure(figsize=(16,16))
plt.imshow(word_cloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```



In [26]:

```
from sklearn.preprocessing import LabelEncoder
```

In [27]:

```
data1 = ['Category']
le = LabelEncoder()
for i in data1:
    data[i] = le.fit_transform(data[i])
```

In [28]:

```
data.head()
```

Out[28]:

	Category	Resume	cleaned_resume
0	6	Skills * Programming Languages: Python (pandas...	Skills Programming Languages Python pandas num...
1	6	Education Details \r\nMay 2013 to May 2017 B.E...	Education Details May 2013 to May 2017 B E UIT...
2	6	Areas of Interest Deep Learning, Control Syste...	Areas of Interest Deep Learning Control System...
3	6	Skills â¢ R â¢ Python â¢ SAP HANA â¢ Table...	Skills R Python SAP HANA Tableau SAP HANA SQL ...
4	6	Education Details \r\n MCA YMCAUST, Faridab...	Education Details MCA YMCAUST Faridabad Haryan...

In [29]:

```
data.tail()
```

Out[29]:

	Category	Resume	cleaned_resume
957	23	Computer Skills: â¢ Proficient in MS office (...	Computer Skills Proficient in MS office Word B...
958	23	â- Willingness to accept the challenges. â- ...	Willingness to a ept the challenges Positive ...
959	23	PERSONAL SKILLS â¢ Quick learner, â¢ Eagerne...	PERSONAL SKILLS Quick learner Eagerness to lea...
960	23	COMPUTER SKILLS & SOFTWARE KNOWLEDGE MS-Power ...	COMPUTER SKILLS SOFTWARE KNOWLEDGE MS Power Po...
961	23	Skill Set OS Windows XP/7/8/8.1/10 Database MY...	Skill Set OS Windows XP 7 8 8 1 10 Database MY...

In [31]:

```
data.Category.value_counts().head()
```

Out[31]:

```
15    84
23    70
8     55
20    48
24    45
Name: Category, dtype: int64
```

In [32]:

```
data.Category.value_counts().tail()
```

Out[32]:

```
2     26
17    25
21    24
5     24
0     20
Name: Category, dtype: int64
```

In [35]:

```
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from scipy.sparse import hstack
```

In [36]:

```
requiredText = data['cleaned_resume'].values
requiredTarget = data['Category'].values
word_vectorizer = TfidfVectorizer(sublinear_tf=True,
                                  stop_words='english')
word_vectorizer.fit(requiredText)
WordFeatures = word_vectorizer.transform(requiredText)
```

In [37]:

```
X_train,X_test,y_train,y_test = train_test_split(WordFeatures,
                                                 requiredTarget,
                                                 random_state=1,
                                                 test_size=0.2,
                                                 shuffle=True,
                                                 stratify=requiredTarget)
print(X_train.shape)
print(X_test.shape)
```

```
(769, 7351)
(193, 7351)
```

In [38]:

```
from sklearn.naive_bayes import MultinomialNB
from sklearn.multiclass import OneVsRestClassifier
from sklearn import metrics
from sklearn.metrics import accuracy_score
from pandas.plotting import scatter_matrix
from sklearn.neighbors import KNeighborsClassifier
from sklearn import metrics
```

In [39]:

```
clf = OneVsRestClassifier(KNeighborsClassifier())
clf.fit(X_train, y_train)
prediction = clf.predict(X_test)
```

In [42]:

```
print('KNC Accuracy Training Data: {:.2f}'.format(clf.score(X_train, y_train)))
print('KNC Accuracy Test Data: {:.2f}'.format(clf.score(X_test, y_test)))
```

```
KNC Accuracy Training Data: 0.99
KNC Accuracy Test Data: 0.99
```

In [43]:

```
print("\n Classification report for classifier %s:\n%s\n" % (clf,
                                                            metrics.classification_repo
```

```
 Classification report for classifier OneVsRestClassifier(estimator=KNeigh
borsClassifier()):
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         4
           1       1.00      1.00      1.00         7
           2       1.00      1.00      1.00         5
           3       1.00      1.00      1.00         8
           4       1.00      1.00      1.00         5
           5       1.00      1.00      1.00         5
           6       1.00      1.00      1.00         8
           7       1.00      1.00      1.00         7
           8       1.00      0.91      0.95        11
           9       0.86      1.00      0.92         6
          10       1.00      1.00      1.00         8
          11       1.00      1.00      1.00         6
          12       1.00      1.00      1.00         9
          13       1.00      1.00      1.00         8
          14       1.00      1.00      1.00         6
          15       1.00      1.00      1.00        17
          16       1.00      1.00      1.00         8
          17       1.00      1.00      1.00         5
          18       1.00      1.00      1.00         8
          19       1.00      1.00      1.00         6
          20       1.00      1.00      1.00        10
          21       1.00      1.00      1.00         5
          22       1.00      1.00      1.00         8
          23       1.00      1.00      1.00        14
          24       1.00      1.00      1.00         9

    accuracy                           0.99       193
   macro avg       0.99      1.00      1.00       193
weighted avg       1.00      0.99      0.99       193
```