# import libraries

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

# import dataset

In [2]:
```
dataset = pd.read_csv('bank.csv')
```

In [3]:
```
dataset.head()
```

Out[3]:

| | age | job | marital | education | default | balance | housing | loan | contact | day | month | duratic |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 59 | admin. | married | secondary | no | 2343 | yes | no | unknown | 5 | may | 10 |
| **1** | 56 | admin. | married | secondary | no | 45 | no | no | unknown | 5 | may | 14 |
| **2** | 41 | technician | married | secondary | no | 1270 | yes | no | unknown | 5 | may | 13 |
| **3** | 55 | services | married | secondary | no | 2476 | yes | no | unknown | 5 | may | 5 |
| **4** | 54 | admin. | married | tertiary | no | 184 | no | no | unknown | 5 | may | 6 |

# check the dataset structe

In [4]:
```
dataset.info
```

Out[4]:
```
<bound method DataFrame.info of        age          job  marital  education default  bal
ance housing loan  \
0       59       admin.  married  secondary      no     2343     yes   no
1       56       admin.  married  secondary      no       45      no   no
2       41   technician  married  secondary      no     1270     yes   no
3       55     services  married  secondary      no     2476     yes   no
4       54       admin.  married   tertiary      no      184      no   no
...    ...          ...      ...        ...     ...      ...     ...  ...
11157   33  blue-collar   single    primary      no        1     yes   no
11158   39     services  married  secondary      no      733      no   no
11159   32   technician   single  secondary      no       29      no   no
11160   43   technician  married  secondary      no        0      no  yes
11161   34   technician  married  secondary      no        0      no   no

        contact  day month  duration  campaign  pdays  previous poutcome  \
0       unknown    5   may      1042         1     -1         0  unknown
1       unknown    5   may      1467         1     -1         0  unknown
2       unknown    5   may      1389         1     -1         0  unknown
3       unknown    5   may       579         1     -1         0  unknown
```

```
4       unknown    5   may       673         2     -1         0  unknown
...         ...  ...   ...       ...       ...    ...       ...      ...
11157  cellular   20   apr       257         1     -1         0  unknown
11158   unknown   16   jun        83         4     -1         0  unknown
11159  cellular   19   aug       156         2     -1         0  unknown
11160  cellular    8   may         9         2    172         5  failure
11161  cellular    9   jul       628         1     -1         0  unknown

      deposit
0         yes
1         yes
2         yes
3         yes
4         yes
...       ...
11157      no
11158      no
11159      no
11160      no
11161      no

[11162 rows x 17 columns]>
```

# find number of rows and column

In [5]:
```
dataset.shape
```

Out[5]:
```
(11162, 17)
```

# describe dataset numerical columns

In [6]:
```
dataset.describe()
```

Out[6]:

| | age | balance | day | duration | campaign | pdays | previou |
|---|---|---|---|---|---|---|---|
| **count** | 11162.000000 | 11162.000000 | 11162.000000 | 11162.000000 | 11162.000000 | 11162.000000 | 11162.00000 |
| **mean** | 41.231948 | 1528.538524 | 15.658036 | 371.993818 | 2.508421 | 51.330407 | 0.83255 |
| **std** | 11.913369 | 3225.413326 | 8.420740 | 347.128386 | 2.722077 | 108.758282 | 2.29200 |
| **min** | 18.000000 | -6847.000000 | 1.000000 | 2.000000 | 1.000000 | -1.000000 | 0.00000 |
| **25%** | 32.000000 | 122.000000 | 8.000000 | 138.000000 | 1.000000 | -1.000000 | 0.00000 |
| **50%** | 39.000000 | 550.000000 | 15.000000 | 255.000000 | 2.000000 | -1.000000 | 0.00000 |
| **75%** | 49.000000 | 1708.000000 | 22.000000 | 496.000000 | 3.000000 | 20.750000 | 1.00000 |
| **max** | 95.000000 | 81204.000000 | 31.000000 | 3881.000000 | 63.000000 | 854.000000 | 58.00000 |

# find the unique values from categorical

# features

In [7]:
```
for col in dataset.select_dtypes(include='object').columns:
    print(col)
    print(dataset[col].unique())
```

```
job
['admin.' 'technician' 'services' 'management' 'retired' 'blue-collar'
 'unemployed' 'entrepreneur' 'housemaid' 'unknown' 'self-employed'
 'student']
marital
['married' 'single' 'divorced']
education
['secondary' 'tertiary' 'primary' 'unknown']
default
['no' 'yes']
housing
['yes' 'no']
loan
['no' 'yes']
contact
['unknown' 'cellular' 'telephone']
month
['may' 'jun' 'jul' 'aug' 'oct' 'nov' 'dec' 'jan' 'feb' 'mar' 'apr' 'sep']
poutcome
['unknown' 'other' 'failure' 'success']
deposit
['yes' 'no']
```

# Find Missing Values

In [8]:
```
features_na = [features for features in dataset.columns if dataset[features].isnull().s
for feature in features_na:
    print(feature, np.round(dataset[feature].isnull().mean(), 4),  ' % missing values')
else:
    print("No missing value found")
```

```
No missing value found
```

# Find Features with One Value

In [9]:
```
for column in dataset.columns:
    print(column,dataset[column].nunique())
```

```
age 76
job 12
marital 3
education 4
default 2
balance 3805
housing 2
loan 2
contact 3
day 31
```

```
month 12
duration 1428
campaign 36
pdays 472
previous 34
poutcome 4
deposit 2
```

# Categorical Features

In [10]:
```
categorical_features=[feature for feature in dataset.columns if ((dataset[feature].dtyp
categorical_features
```

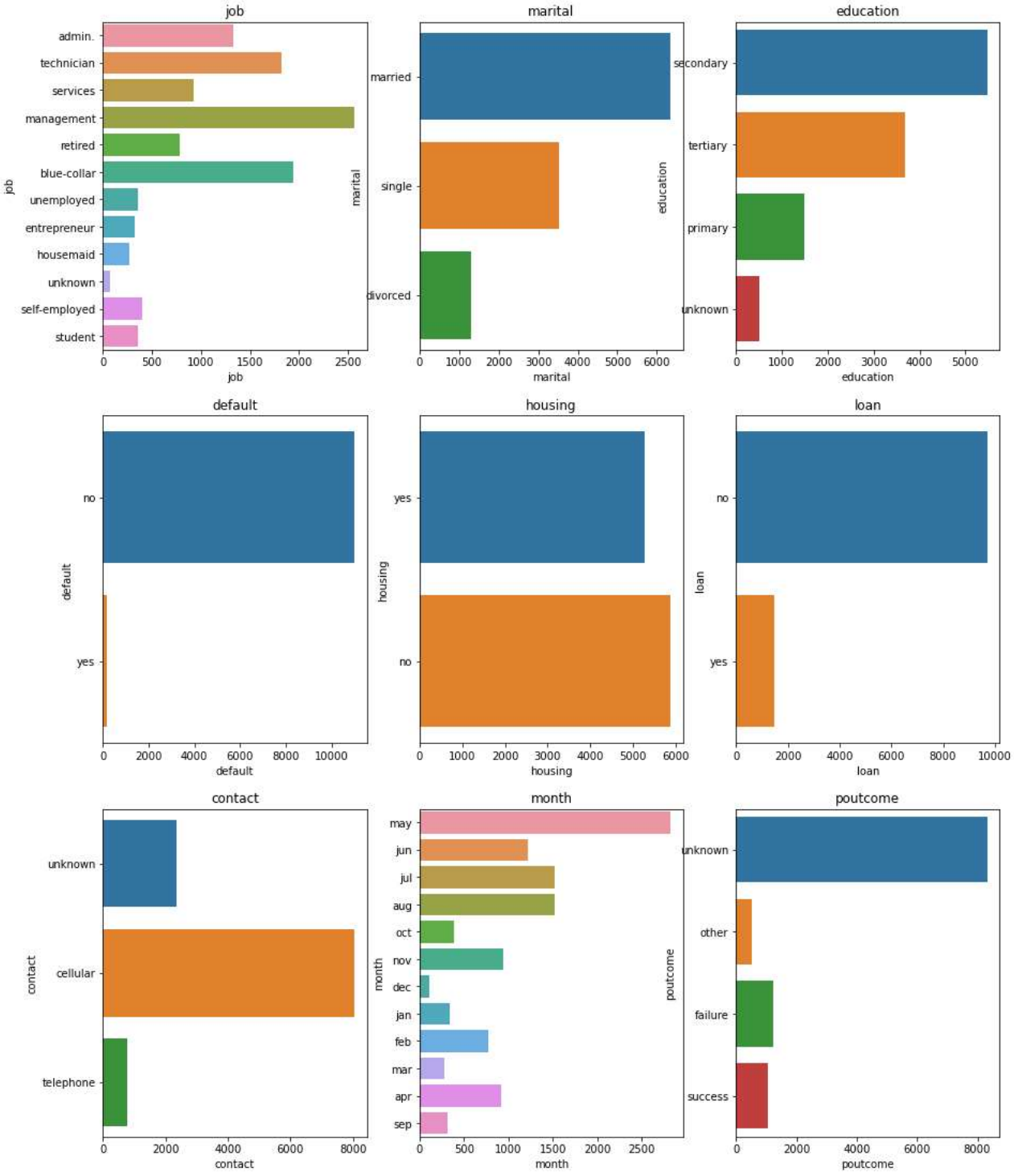
Out[10]:
```
['job',
 'marital',
 'education',
 'default',
 'housing',
 'loan',
 'contact',
 'month',
 'poutcome']
```

In [11]:
```
for feature in categorical_features:
    print('The feature is {} and number of categories are {}'.format(feature,len(datase
```

```
The feature is job and number of categories are 12
The feature is marital and number of categories are 3
The feature is education and number of categories are 4
The feature is default and number of categories are 2
The feature is housing and number of categories are 2
The feature is loan and number of categories are 2
The feature is contact and number of categories are 3
The feature is month and number of categories are 12
The feature is poutcome and number of categories are 4
```

# check count based on categorical features

In [12]:
```
plt.figure(figsize=(15,80), facecolor='white')
plotnumber =1
for categorical_feature in categorical_features:
    ax = plt.subplot(12,3,plotnumber)
    sns.countplot(y=categorical_feature,data=dataset)
    plt.xlabel(categorical_feature)
    plt.title(categorical_feature)
    plotnumber+=1
plt.show()
```




# Explore the Numerical Features

In [13]:
```
# list of numerical variables
numerical_features = [feature for feature in dataset.columns if ((dataset[feature].dtyp
print('Number of numerical variables: ', len(numerical_features))

# visualise the numerical variables
dataset[numerical_features].head()
```

```
Number of numerical variables:  7
```

Out[13]:

| | age | balance | day | duration | campaign | pdays | previous |
|---|---|---|---|---|---|---|---|

| | age | balance | day | duration | campaign | pdays | previous |
|---|---|---|---|---|---|---|---|
| **0** | 59 | 2343 | 5 | 1042 | 1 | -1 | 0 |
| **1** | 56 | 45 | 5 | 1467 | 1 | -1 | 0 |
| **2** | 41 | 1270 | 5 | 1389 | 1 | -1 | 0 |
| **3** | 55 | 2476 | 5 | 579 | 1 | -1 | 0 |
| **4** | 54 | 184 | 5 | 673 | 2 | -1 | 0 |