In [1234]:
```
import pandas as pd
import numpy as np

hc = pd.read_csv('Headcount.csv', parse_dates=[0], index_col =0)
nh = pd.read_csv('New_Hire.csv', parse_dates=[0], index_col =0)
```

Headcount

In [1235]:
```
#remove white space
hc.rename(columns=lambda x: x.replace(" ", "_"), inplace=True)
nh.rename(columns=lambda x: x.replace(" ", "_"), inplace=True)
```

In [1236]:
```
hc.head()
```

Out[1236]:

	Manager_ID	Geography	Region	Country_ID	Work_Location_Name	Sex_Code	Hire_Date	Salary_I
Month ID								
2020-01-01	Management	AP	AU/NZ	Australia	NORWICH	Female	May 1, 2003	
2020-01-01	Non-Management	AP	AU/NZ	Australia	Not Available	Male	Nov 16, 2016	
2020-01-01	Non-Management	AP	AU/NZ	Australia	SOUTHGATE	Male	Jul 1, 2013	
2020-01-01	Non-Management	AP	AU/NZ	Australia	NORWICH	Male	Jun 15, 1981	
2020-01-01	Management	AP	ASEAN	Malaysia	SELANGOR	Male	May 1, 2007	

In [1237]:
```
hc.dtypes
```

Out[1237]:
```
Manager_ID                    object
Geography                     object
Region                        object
Country_ID                    object
Work_Location_Name            object
Sex_Code                      object
Hire_Date                     object
Salary_Band_Code              object
Salary_Band_Date              object
Local_Position_Date           object
Service_Group                 object
Segment_Name                  object
Sec_Job_Category_Name         object
Pri_Job_Category_Name         object
Sec_Job_Category_Code         object
Pri_Job_Category_Code         object
Primary_Job_Role              object
Role_/_Specialty              object
Specialty                     object
Is_Distinguished_Engineer     object
Is_Fellow                     object
Tech_ID                       object
Is_Tech_Job_Role              object
URM                           object
Diversity_(grouping)          object
Diversity                     object
Level_4_Name                  object
Level_5_Name                  object
Level_6_Name                  object
Level_7_Name                  object
Level_8_Name                  object
Level_9_Name                  object
Level_10_Name                 object
Hire_Type                     object
Is_CIC                        object
Salary_Band_Movement_Up      float64
Years_In_Band                float64
Years_In_Previous_Band       float64
Empl_Count                     int64
Grouping_Code_1               object
Grouping_Code_2               object
Hire_Year                      int64
Salary_Band_Year              object
dtype: object
```

In [1238]:

```
nh.head()
```

Out[1238]:

	Country	Geography	Region	Work_Location_Code	Work_Location_Name	Month_ID	BP_CNUM	Status
Row								
1	USA	NaN	US	???	Not Available	2020-January	5G9305897	A
4	USA	NaN	US	BNT	AUSTIN	2020-January	3J2886897	A
16	USA	NaN	US	CF5	IBM CLOUD-DAL11 (USSL1783)	2020-January	3J3641897	A
17	USA	NaN	US	CF5	IBM CLOUD-DAL11 (USSL1783)	2020-January	4J6270897	A
33	USA	NaN	US	HK9	EMERYVILLE-ASPERA	2020-January	4J5897897	A

In [1239]:

```
nh.dtypes
```

Out[1239]:

```
Country                  object
Geography                object
Region                   object
Work_Location_Code       object
Work_Location_Name       object
Month_ID                 object
BP_CNUM                  object
Status                   object
URM                      object
Sex_Short_ID             object
Is_Tech_Job_Role         object
Sec_Job_Category_Code    object
Primary_Job_Role         object
Pri_Job_Category_Name    object
Sec_Job_Category_Name    object
Salary_Band_Code          int64
Movement_Group_1_ID      object
Movement_Group_2_ID      object
Grouping_Code_1          object
Grouping_Code_2          object
dtype: object
```

In [1240]:

```
nh['Status'].unique().tolist()
```

Out[1240]:

```
['A']
```

In [1241]:

```
nh = nh.drop(['Status'], axis=1)
```

In [1242]:
```
nh.dtypes
```

Out[1242]:
```
Country                   object
Geography                 object
Region                    object
Work_Location_Code        object
Work_Location_Name        object
Month_ID                  object
BP_CNUM                   object
URM                       object
Sex_Short_ID              object
Is_Tech_Job_Role          object
Sec_Job_Category_Code     object
Primary_Job_Role          object
Pri_Job_Category_Name     object
Sec_Job_Category_Name     object
Salary_Band_Code           int64
Movement_Group_1_ID       object
Movement_Group_2_ID       object
Grouping_Code_1           object
Grouping_Code_2           object
dtype: object
```

In [1243]:
```
hc.describe(include='all')
```

Out[1243]:

	Manager_ID	Geography	Region	Country_ID	Work_Location_Name	Sex_Code	Hire_Date	Salary_Band
count	8126	3608	8126	8126	8126	8126	8126	
unique	2	6	20	55	303	2	2840	
top	Non-Management	Europe	US	USA	AUSTIN	Male	Jan 1, 2014	
freq	7464	1981	3975	3975	842	6422	190	
mean	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
std	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
min	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
25%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
50%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
75%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
max	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

Data Cleaning

Join tables

In [1244]:
```
df = pd.concat([hc,nh], ignore_index=True, sort=False)
```

In [1245]:
```
#another way to combine Country and Country_ID
#df['Country'] = np.where(df['Country'].isna(), df['Country_ID'], df['Country'])
```

In [1246]:
```
df.head()
```

Out[1246]:

	Manager_ID	Geography	Region	Country_ID	Work_Location_Name	Sex_Code	Hire_Date	Salary_Band_Cod
0	Management	AP	AU/NZ	Australia	NORWICH	Female	May 1, 2003	1
1	Non-Management	AP	AU/NZ	Australia	Not Available	Male	Nov 16, 2016	1
2	Non-Management	AP	AU/NZ	Australia	SOUTHGATE	Male	Jul 1, 2013	1
3	Non-Management	AP	AU/NZ	Australia	NORWICH	Male	Jun 15, 1981	1
4	Management	AP	ASEAN	Malaysia	SELANGOR	Male	May 1, 2007	1

In [1247]:
```
df = df.rename(columns={"Diversity_(grouping)":"Diversity_Grouping"})
```

In [1248]:
```
df.head()
```

Out[1248]:

	Manager_ID	Geography	Region	Country_ID	Work_Location_Name	Sex_Code	Hire_Date	Salary_Band_Cod
0	Management	AP	AU/NZ	Australia	NORWICH	Female	May 1, 2003	1
1	Non-Management	AP	AU/NZ	Australia	Not Available	Male	Nov 16, 2016	1
2	Non-Management	AP	AU/NZ	Australia	SOUTHGATE	Male	Jul 1, 2013	1
3	Non-Management	AP	AU/NZ	Australia	NORWICH	Male	Jun 15, 1981	1
4	Management	AP	ASEAN	Malaysia	SELANGOR	Male	May 1, 2007	1

In [1249]:
```
df['Country']=df.Country.combine_first(df.Country_ID)
```

In [1250]:
```
#drop Country_ID
df = df.drop(['Country_ID'], axis=1)
```

In [1251]:
```
#drop Region, since it is redundant from COUNTRY
df = df.drop(['Region'], axis=1)
```

In [1252]:
```
#combine into SEX column
df['Sex']=df.Sex_Code.combine_first(df.Sex_Short_ID)
```

In [1253]:

```
#drop old Sex columns
df = df.drop(['Sex_Code'], axis=1)
df = df.drop(['Sex_Short_ID'], axis=1)
```

In [1254]:

```
#convert 'Hire Date' to month/year
df['Hire_Date'] = pd.to_datetime(df['Hire_Date']).dt.strftime('%m/%Y')
```

In [1255]:

```
#combine 'Hire_Date' and 'Month_ID'
df['Hire_Date']=df.Hire_Date.combine_first(df.Month_ID)
```

In [1256]:

```
#drop Month_ID
df = df.drop(['Month_ID'], axis=1)
```

In [1257]:

```
#rename 'Movement Group 2 ID' to Hire_Type
df['Hire_Type']=df.Hire_Type.combine_first(df.Movement_Group_2_ID)
```

In [1258]:

```
#drop old Movement Group 2 ID column
df = df.drop(['Movement_Group_2_ID'], axis=1)
```

In [1259]:

```
#remove 'Salary_Band_Movement_Up' column due to value only being 1 or NaN
df = df.drop(['Salary_Band_Movement_Up'], axis=1)
```

In [1260]:

```
#drop Diversity_grouping column since it is represented by Sex and URM
df = df.drop(['Diversity_Grouping'], axis=1)
```

In [1261]:

```
df.head()
```

Out[1261]:

	Manager_ID	Geography	Work_Location_Name	Hire_Date	Salary_Band_Code	Salary_Band_Date	Local_Posi
0	Management	AP	NORWICH	05/2003	10	Jul 1, 2009	O
1	Non-Management	AP	Not Available	11/2016	10	Nov 16, 2016	Au
2	Non-Management	AP	SOUTHGATE	07/2013	10	Jun 1, 2018	Se
3	Non-Management	AP	NORWICH	06/1981	10	May 1, 2000	O
4	Management	AP	SELANGOR	05/2007	10	Nov 1, 2014	M

In [1262]:

```python
#clean null values
def assess_NA(data):
    """
    Returns a pandas dataframe denoting the total number of NA values and the percentage of NA values in each column.
    The column names are noted on the index.

    Parameters
    ----------
    data: dataframe
    """
    # pandas series denoting features and the sum of their null values
    null_sum = data.isnull().sum()# instantiate columns for missing data
    total = null_sum.sort_values(ascending=False)
    percent = ( ((null_sum / len(data.index))*100).round(2) ).sort_values(ascending=False)

    # concatenate along the columns to create the complete dataframe
    df_NA = pd.concat([total, percent], axis=1, keys=['Number of NA', 'Percent NA'])

    # drop rows that don't have any missing data; omit if you want to keep all rows
    df_NA = df_NA[ (df_NA.T != 0).any() ]

    return df_NA
```

In [1263]:

```
#show frequency and percentage of nulls
df_NA = assess_NA(df)
df_NA
```

Out[1263]:

	Number of NA	Percent NA
Grouping_Code_2	8189	99.94
BP_CNUM	8126	99.17
Work_Location_Code	8126	99.17
Movement_Group_1_ID	8126	99.17
Grouping_Code_1	4857	59.28
Geography	4545	55.47
Years_In_Previous_Band	2298	28.04
Hire_Type	922	11.25
Specialty	243	2.97
Salary_Band_Date	71	0.87
Years_In_Band	71	0.87
Local_Position_Date	70	0.85
Role_/_Specialty	68	0.83
Is_Fellow	68	0.83
Is_Distinguished_Engineer	68	0.83
Diversity	68	0.83
Pri_Job_Category_Code	68	0.83
Segment_Name	68	0.83
Service_Group	68	0.83
Tech_ID	68	0.83
Manager_ID	68	0.83
Level_4_Name	68	0.83
Salary_Band_Year	68	0.83
Level_5_Name	68	0.83
Level_6_Name	68	0.83
Level_7_Name	68	0.83
Level_8_Name	68	0.83
Level_9_Name	68	0.83
Level_10_Name	68	0.83
Is_CIC	68	0.83
Empl_Count	68	0.83
Hire_Year	68	0.83
Primary_Job_Role	8	0.10

In [1264]:

```
#should drop columns:'Grouping_Code_2', 'Movement_Group_1_ID', 'Status', 'BP_CNUM
', 'Work_Location_Code'
df = df.drop(['Grouping_Code_2','Movement_Group_1_ID','BP_CNUM','Work_Location_Co
de'], axis=1)
```

In [1265]:

```
df.head()
```

Out[1265]:

	Manager_ID	Geography	Work_Location_Name	Hire_Date	Salary_Band_Code	Salary_Band_Date	Local_Posi
0	Management	AP	NORWICH	05/2003	10	Jul 1, 2009	O
1	Non-Management	AP	Not Available	11/2016	10	Nov 16, 2016	Auį
2	Non-Management	AP	SOUTHGATE	07/2013	10	Jun 1, 2018	Sep
3	Non-Management	AP	NORWICH	06/1981	10	May 1, 2000	O
4	Management	AP	SELANGOR	05/2007	10	Nov 1, 2014	M

In [1266]:

```
df_NA = assess_NA(df)
df_NA
```

Out[1266]:

	Number of NA	Percent NA
Grouping_Code_1	4857	59.28
Geography	4545	55.47
Years_In_Previous_Band	2298	28.04
Hire_Type	922	11.25
Specialty	243	2.97
Salary_Band_Date	71	0.87
Years_In_Band	71	0.87
Local_Position_Date	70	0.85
Manager_ID	68	0.83
Segment_Name	68	0.83
Pri_Job_Category_Code	68	0.83
Role_/_Specialty	68	0.83
Is_Distinguished_Engineer	68	0.83
Is_Fellow	68	0.83
Tech_ID	68	0.83
Diversity	68	0.83
Level_4_Name	68	0.83
Level_5_Name	68	0.83
Level_6_Name	68	0.83
Level_7_Name	68	0.83
Level_8_Name	68	0.83
Level_9_Name	68	0.83
Level_10_Name	68	0.83
Is_CIC	68	0.83
Empl_Count	68	0.83
Hire_Year	68	0.83
Salary_Band_Year	68	0.83
Service_Group	68	0.83
Primary_Job_Role	8	0.10

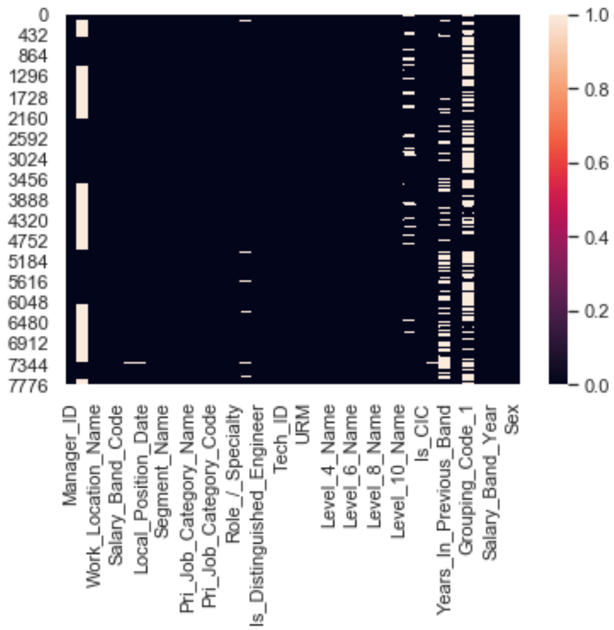
Data Exploration

In [1267]:

```
import matplotlib.pyplot as plt
plt.rc("font", size=14)
import seaborn as sns
sns.set(style="white")
sns.set(style="whitegrid", color_codes=True)
```

In [1268]:
```
sns.heatmap(df.isnull())
```

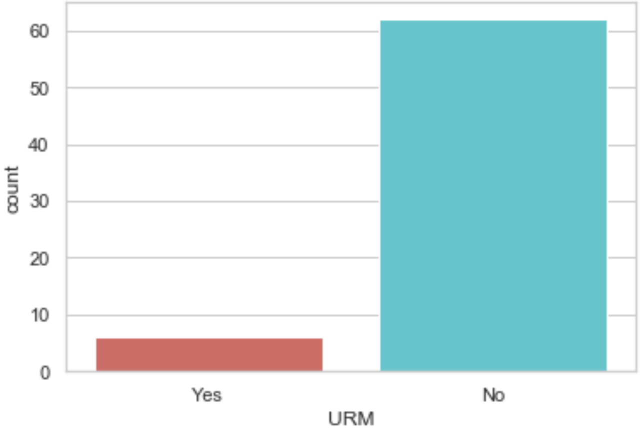
Out[1268]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff9f702c510>
```



URM

New Hire

In [1269]:
```
sns.countplot(x='URM', data=nh, palette='hls')
plt.show()
```

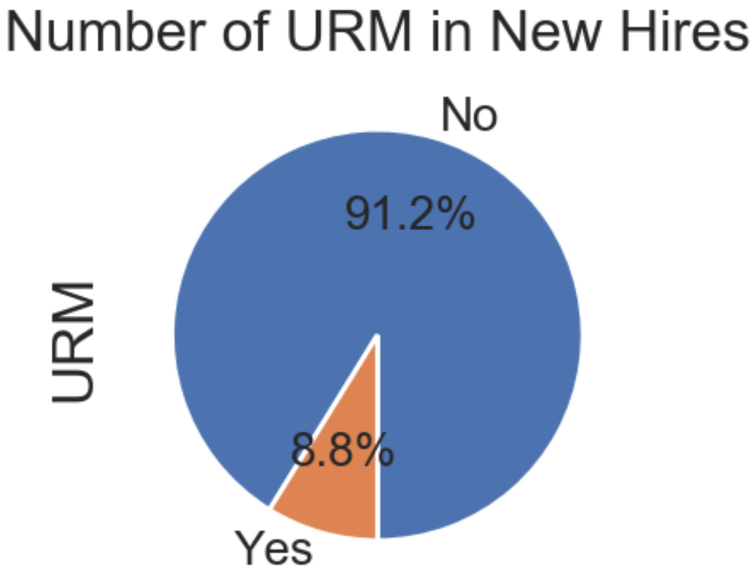


In [1270]:
```
nh.URM.value_counts()
```

Out[1270]:
```
No      62
Yes      6
Name: URM, dtype: int64
```

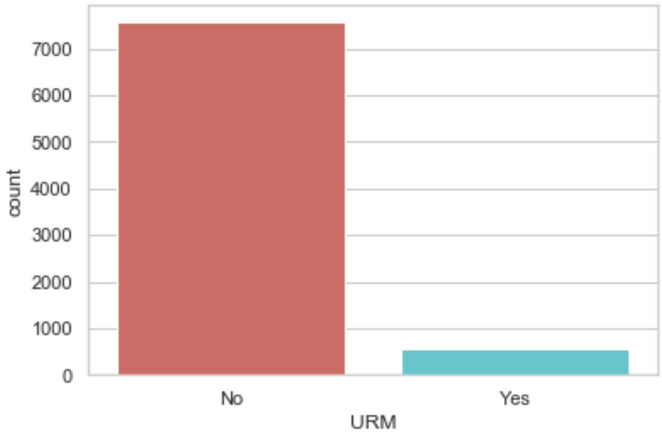
In [1271]:
```
fig = plt.figure(figsize=(2,2), dpi=200)
ax = plt.subplot(111)

nh.URM.value_counts().plot(kind='pie',ax=ax, autopct='%1.1f%%',startangle=270, fo
ntsize=10)
plt.title('Number of URM in New Hires')
```

Out[1271]:
```
Text(0.5, 1.0, 'Number of URM in New Hires')
```



Head Count

In [1272]:
```
sns.countplot(x='URM', data=hc, palette='hls')
plt.show()
```



In [1273]:
```
hc.URM.value_counts()
```

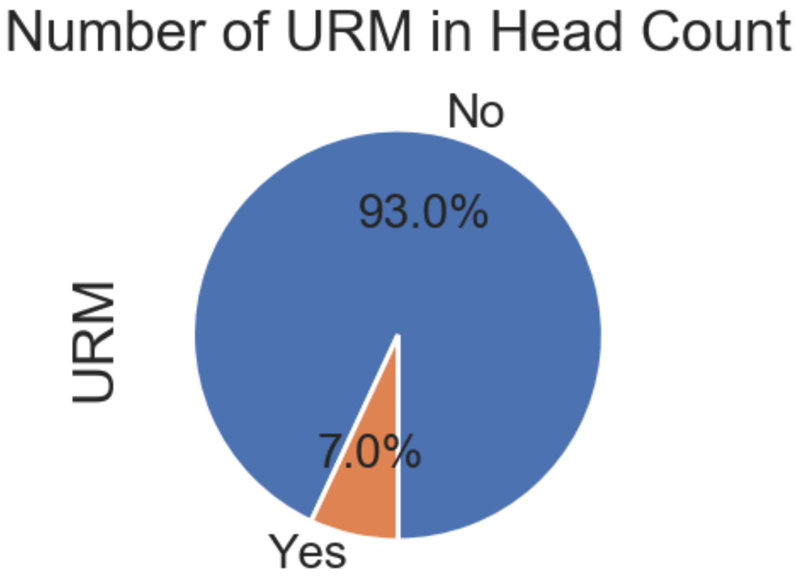
Out[1273]:
```
No      7561
Yes      565
Name: URM, dtype: int64
```

In [1274]:

```
fig = plt.figure(figsize=(2,2), dpi=200)
ax = plt.subplot(111)

hc.URM.value_counts().plot(kind='pie',ax=ax, autopct='%1.1f%%',startangle=270, fontsize=10)
plt.title('Number of URM in Head Count')
```

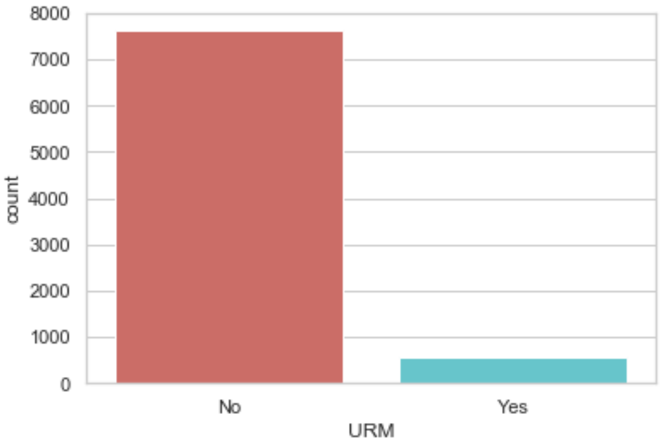
Out[1274]:

```
Text(0.5, 1.0, 'Number of URM in Head Count')
```



In [1275]:

```
sns.countplot(x='URM', data=df, palette='hls')
plt.show()
```



In [1276]:

```
df.URM.value_counts()
```

Out[1276]:

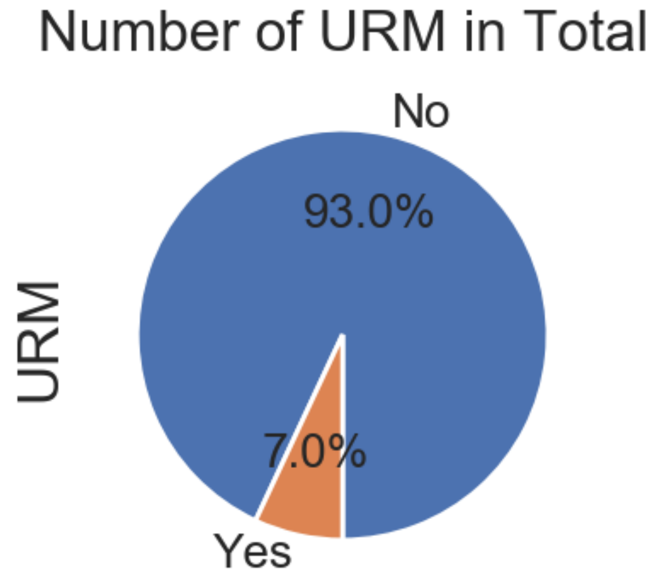
```
No      7623
Yes      571
Name: URM, dtype: int64
```

In [1277]:

```
fig = plt.figure(figsize=(2,2), dpi=200)
ax = plt.subplot(111)

df.URM.value_counts().plot(kind='pie',ax=ax, autopct='%1.1f%%',startangle=270, fo
ntsize=10)
plt.title('Number of URM in Total')
```

Out[1277]:

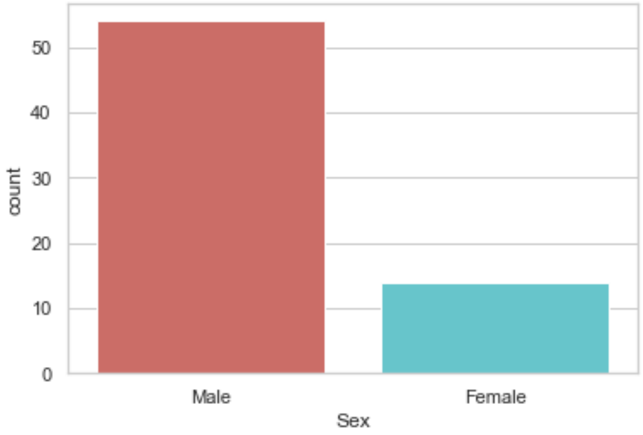
```
Text(0.5, 1.0, 'Number of URM in Total')
```



Sex

In [1278]:

```
sns.countplot(x='Sex_Short_ID', data=nh, palette='hls')
plt.xlabel('Sex')
plt.show()
```



In [1279]:

```
nh.Sex_Short_ID.value_counts()
```

Out[1279]:

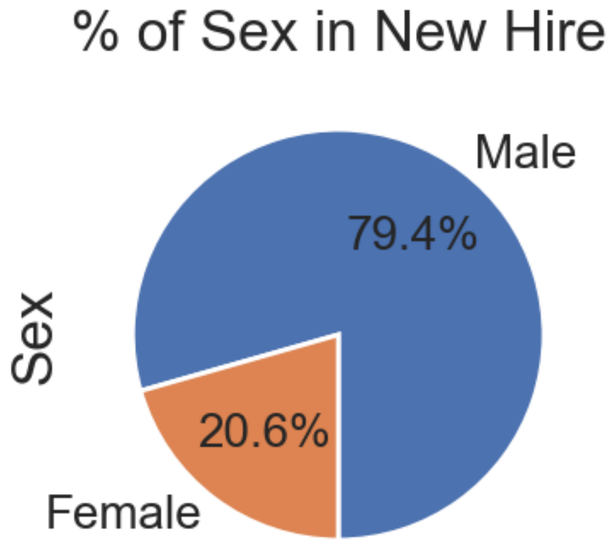
```
Male      54
Female    14
Name: Sex_Short_ID, dtype: int64
```

In [1280]:

```
fig = plt.figure(figsize=(2,2), dpi=200)
ax = plt.subplot(111)

nh.Sex_Short_ID.value_counts().plot(kind='pie',ax=ax, autopct='%1.1f%%',startangl
e=270, fontsize=10)
plt.title('% of Sex in New Hire')
plt.ylabel('Sex')
```

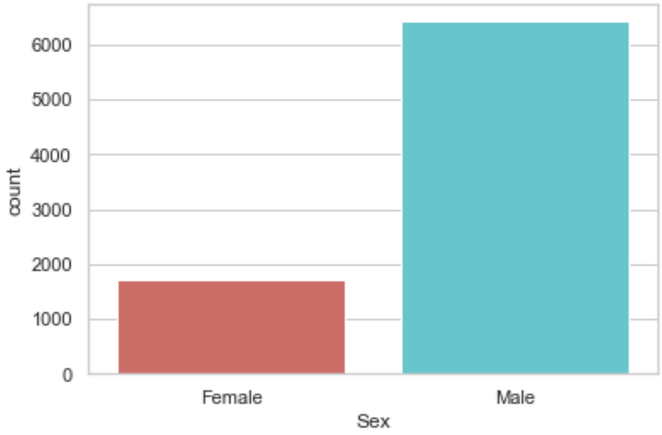
Out[1280]:

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



In [1281]:

```
sns.countplot(x='Sex_Code', data=hc, palette='hls')
plt.xlabel('Sex')
plt.show()
```



In [1282]:

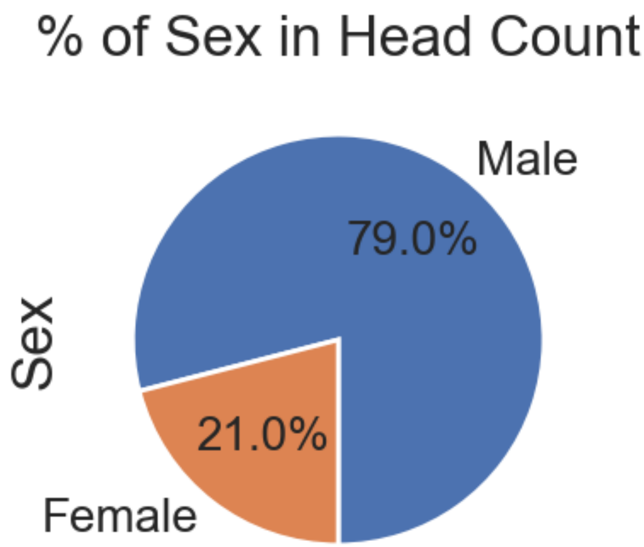
```
hc.Sex_Code.value_counts()
```

Out[1282]:

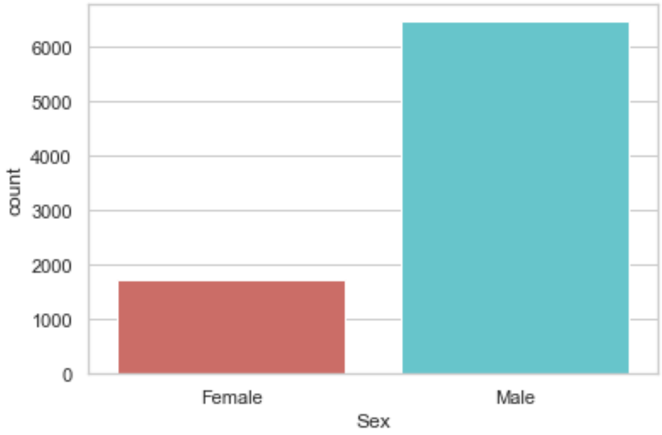
```
Male      6422
Female    1704
Name: Sex_Code, dtype: int64
```

In [1283]:
```
fig = plt.figure(figsize=(2,2), dpi=200)
ax = plt.subplot(111)

hc.Sex_Code.value_counts().plot(kind='pie',ax=ax, autopct='%1.1f%%',startangle=270, fontsize=10)
plt.title('% of Sex in Head Count')
plt.ylabel('Sex')
```

Out[1283]:
```
Text(0, 0.5, 'Sex')
```



In [1284]:
```
sns.countplot(x='Sex', data=df, palette='hls')
plt.xlabel('Sex')
plt.show()
```

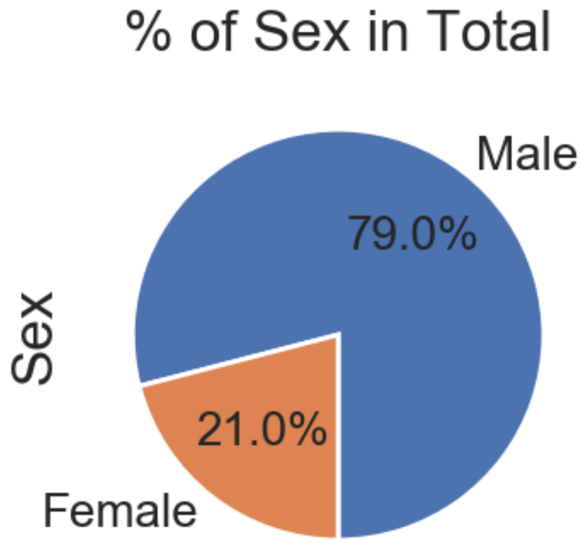


In [1285]:
```
df.Sex.value_counts()
```

Out[1285]:
```
Male      6476
Female    1718
Name: Sex, dtype: int64
```

```
In [1286]: fig = plt.figure(figsize=(2,2), dpi=200)
           ax = plt.subplot(111)

           df.Sex.value_counts().plot(kind='pie',ax=ax, autopct='%1.1f%%',startangle=270, fo
           ntsize=10)
           plt.title('% of Sex in Total')
           plt.ylabel('Sex')
```

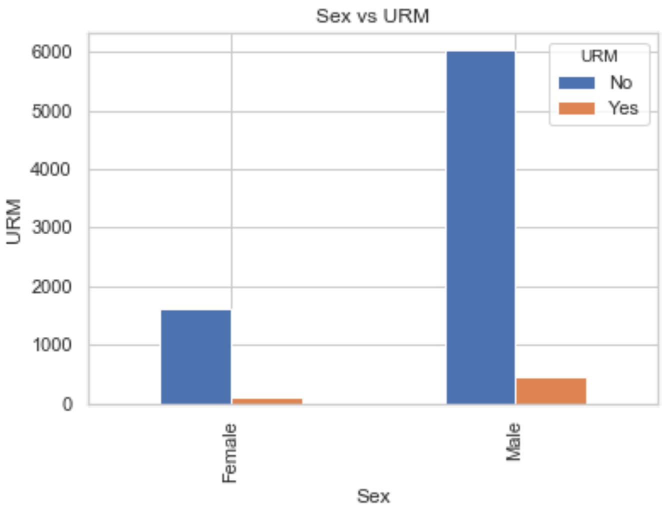
```
Out[1286]: Text(0, 0.5, 'Sex')
```



URM vs Sex

```
In [1287]: %matplotlib inline
           pd.crosstab(df.Sex,df.URM).plot(kind='bar')
           plt.title('Sex vs URM')
           plt.xlabel('Sex')
           plt.ylabel('URM')
```

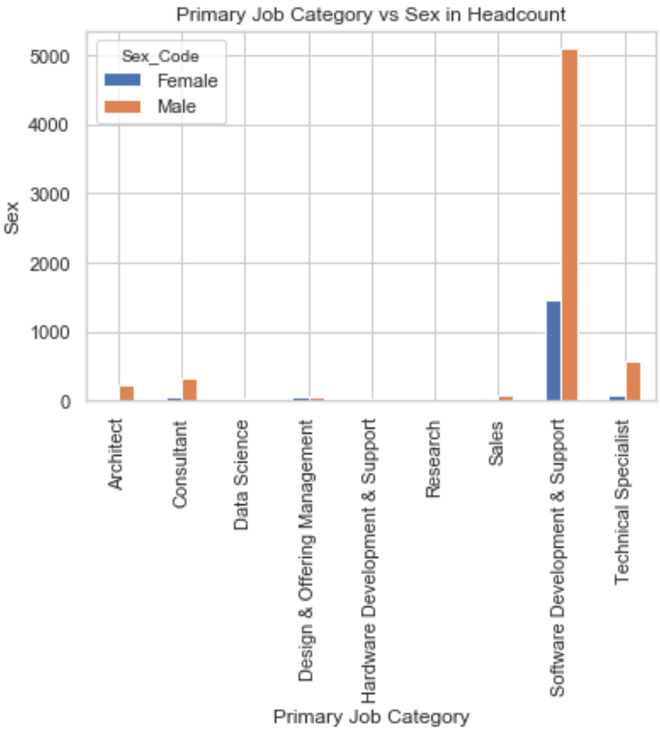
```
Out[1287]: Text(0, 0.5, 'URM')
```



Primary Job Category vs Sex

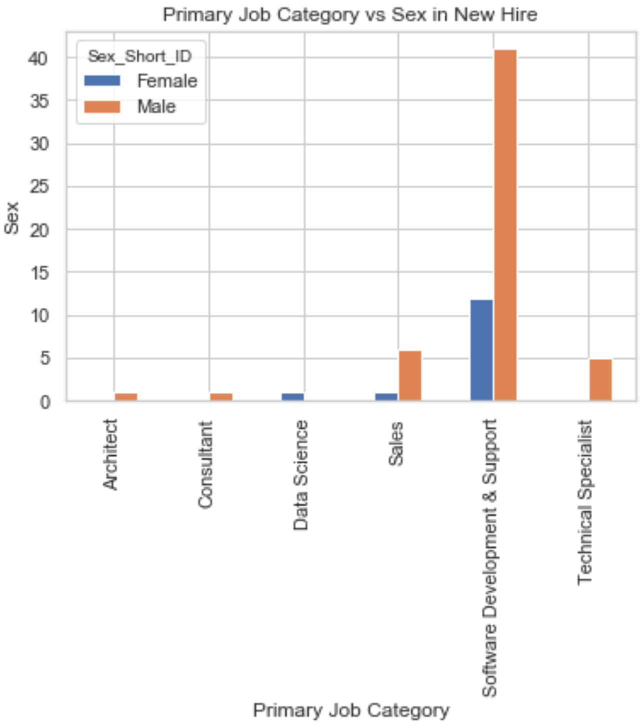
In [1288]:
```
%matplotlib inline
pd.crosstab(hc.Pri_Job_Category_Name,hc.Sex_Code).plot(kind='bar')
plt.title('Primary Job Category vs Sex in Headcount')
plt.xlabel('Primary Job Category')
plt.ylabel('Sex')
```

Out[1288]:
```
Text(0, 0.5, 'Sex')
```



```
In [1289]: %matplotlib inline
           pd.crosstab(nh.Pri_Job_Category_Name,nh.Sex_Short_ID).plot(kind='bar')
           plt.title('Primary Job Category vs Sex in New Hire')
           plt.xlabel('Primary Job Category')
           plt.ylabel('Sex')
```

```
Out[1289]: Text(0, 0.5, 'Sex')
```



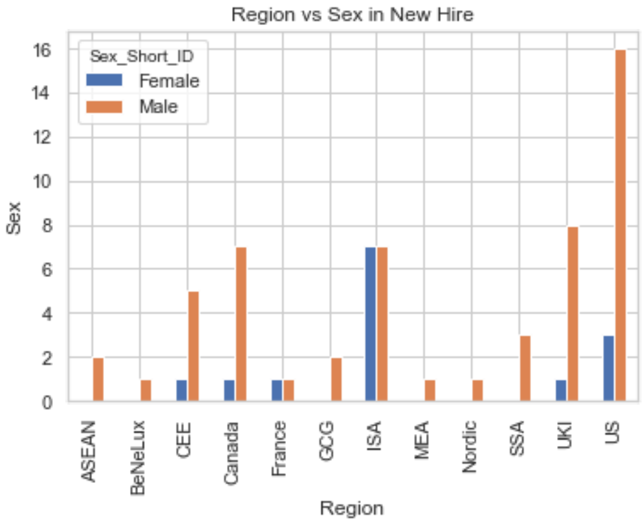
Region vs Sex

In [1290]:

```
#new hire
%matplotlib inline
pd.crosstab(nh.Region,nh.Sex_Short_ID).plot(kind='bar')
plt.title('Region vs Sex in New Hire')
plt.xlabel('Region')
plt.ylabel('Sex')
```

Out[1290]:

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

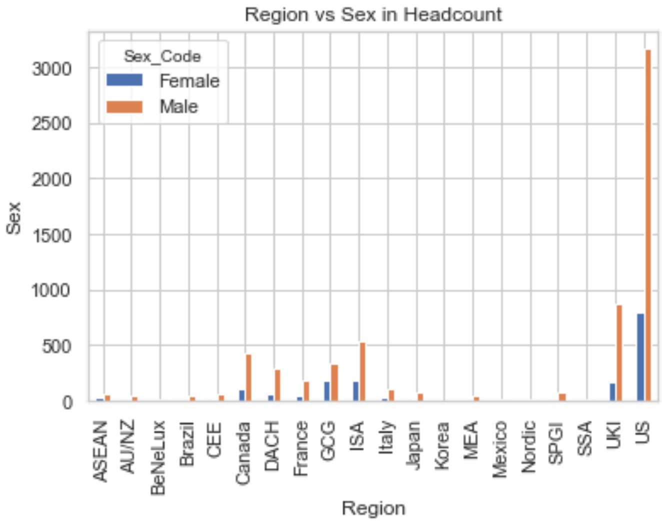


In [1291]:

```
#head count
%matplotlib inline
pd.crosstab(hc.Region,hc.Sex_Code).plot(kind='bar')
plt.title('Region vs Sex in Headcount')
plt.xlabel('Region')
plt.ylabel('Sex')
```

Out[1291]:

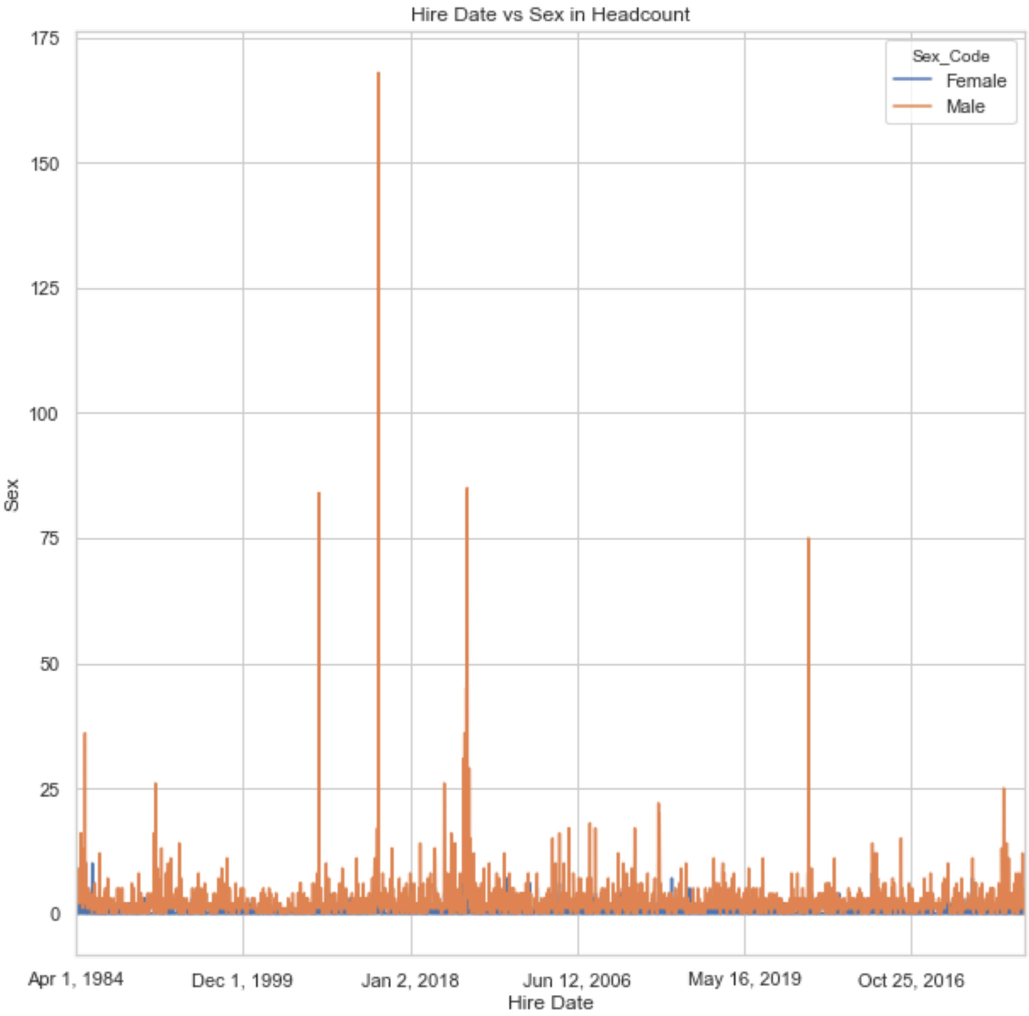
```
Text(0, 0.5, 'Sex')
```



Hire Date vs Sex

```
In [1298]: #head count
           %matplotlib inline
           pd.crosstab(hc.Hire_Date,hc.Sex_Code).plot(kind='line', figsize=(10,10))
           plt.title('Hire Date vs Sex in Headcount')
           plt.xlabel('Hire Date')
           plt.ylabel('Sex')
```

```
Out[1298]: Text(0, 0.5, 'Sex')
```


Hire Date vs Sex in Headcount
Sex_Code
Female
Male
Sex
Apr 1, 1984
Dec 1, 1999
Jan 2, 2018
Jun 12, 2006
May 16, 2019
Oct 25, 2016
Hire Date