PROJECT SUMMARY------ This Project is focused on analyzing a recruitment funnel using candidate data.The goal is to understand which types of candidates are most likely to be interested in changing jobs,based on their experience ,education and current status.We performed exploratory data analysis and visualization key patterns related to job change interest.

KEY INSIGHTS------

Most candidates who want to change jobs have relevant work experience. Job change interest is higher among candidates with a Master's degree. Candidates currently enrolled in Full-time courses are less likely to want a job change. Some cities show a significantly higher interest in job change ,which can guide location-based hiring strategies

RECOMMENDATIONS------

1.Focus hiring efforts on candidates with relevant experience ,as they are more likely to be open to job change. 2.Prioritize outreach to candidates with Master's degrees,who show higher job change interest. 3.Avoid pushing job offers to full-time students unless roles are internships or flexible. 4.Target recruitment campaigns in cities where job change interest is high improve conversion rates.

In [3]:

```
import pandas as pd
df=pd.read_csv('aug_train.csv')
df.head()
```

Out[3]:

	enrollee_id	city	city_development_index	gender	relevent_experience	enrolled_univ
0	8949	city_103	0.920	Male	Has relevent experience	no_enrol
1	29725	city_40	0.776	Male	No relevent experience	no_enrol
2	11561	city_21	0.624	NaN	No relevent experience	Full time c
3	33241	city_115	0.789	NaN	No relevent experience	
4	666	city_162	0.767	Male	Has relevent experience	no_enrol

In [4]:

```
df.columns
```

```
Out[4]: Index(['enrollee_id', 'city', 'city_development_index', 'gender',
               'relevent_experience', 'enrolled_university', 'education_level',
               'major_discipline', 'experience', 'company_size', 'company_type',
               'last_new_job', 'training_hours', 'target'],
              dtype='object')
```

```
In [5]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 19158 entries, 0 to 19157
Data columns (total 14 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   enrollee_id             19158 non-null  int64  
 1   city                    19158 non-null  object 
 2   city_development_index  19158 non-null  float64
 3   gender                  14650 non-null  object 
 4   relevent_experience     19158 non-null  object 
 5   enrolled_university     18772 non-null  object 
 6   education_level         18698 non-null  object 
 7   major_discipline        16345 non-null  object 
 8   experience              19093 non-null  object 
 9   company_size            13220 non-null  object 
 10  company_type            13018 non-null  object 
 11  last_new_job            18735 non-null  object 
 12  training_hours          19158 non-null  int64  
 13  target                  19158 non-null  float64
dtypes: float64(2), int64(2), object(10)
memory usage: 2.0+ MB
```

```
In [6]: df.isnull().sum()
```

```
Out[6]: enrollee_id                  0
        city                         0
        city_development_index       0
        gender                    4508
        relevent_experience          0
        enrolled_university        386
        education_level            460
        major_discipline          2813
        experience                  65
        company_size              5938
        company_type              6140
        last_new_job               423
        training_hours               0
        target                       0
        dtype: int64
```

```
In [7]: df.describe()
```

Out[7]:

	enrollee_id	city_development_index	training_hours	target
count	19158.000000	19158.000000	19158.000000	19158.000000
mean	16875.358179	0.828848	65.366896	0.249348
std	9616.292592	0.123362	60.058462	0.432647
min	1.000000	0.448000	1.000000	0.000000
25%	8554.250000	0.740000	23.000000	0.000000
50%	16982.500000	0.903000	47.000000	0.000000
75%	25169.750000	0.920000	88.000000	0.000000
max	33380.000000	0.949000	336.000000	1.000000

In [8]:

```
df['target'].value_counts()
```

Out[8]:

```
target
0.0    14381
1.0     4777
Name: count, dtype: int64
```

In [9]:

```
df['relevent_experience'].value_counts()
```

Out[9]:

```
relevent_experience
Has relevent experience    13792
No relevent experience      5366
Name: count, dtype: int64
```

In [11]:

```
df['education_level'].value_counts()
```

Out[11]:

```
education_level
Graduate          11598
Masters            4361
High School        2017
Phd                 414
Primary School      308
Name: count, dtype: int64
```

In [12]:

```
df['enrolled_university'].value_counts()
```

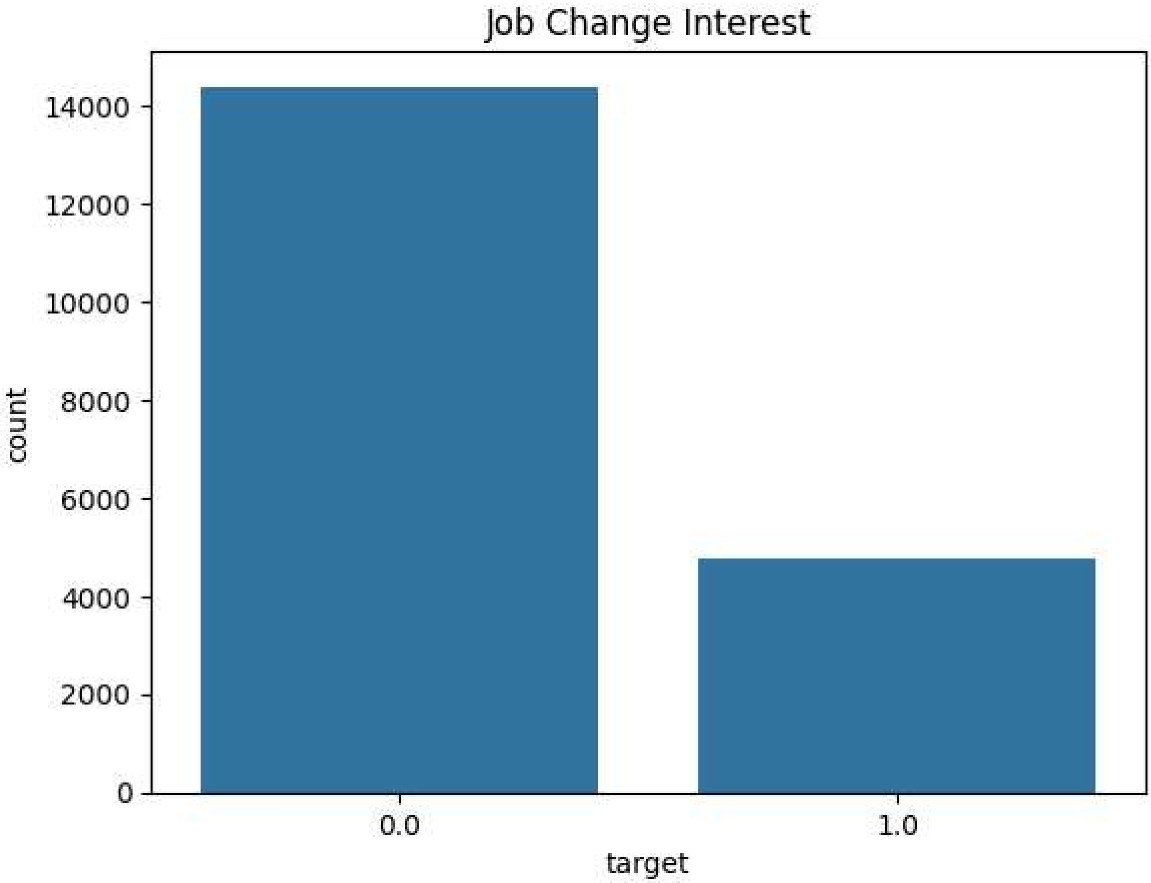
Out[12]:

```
enrolled_university
no_enrollment       13817
Full time course     3757
Part time course     1198
Name: count, dtype: int64
```

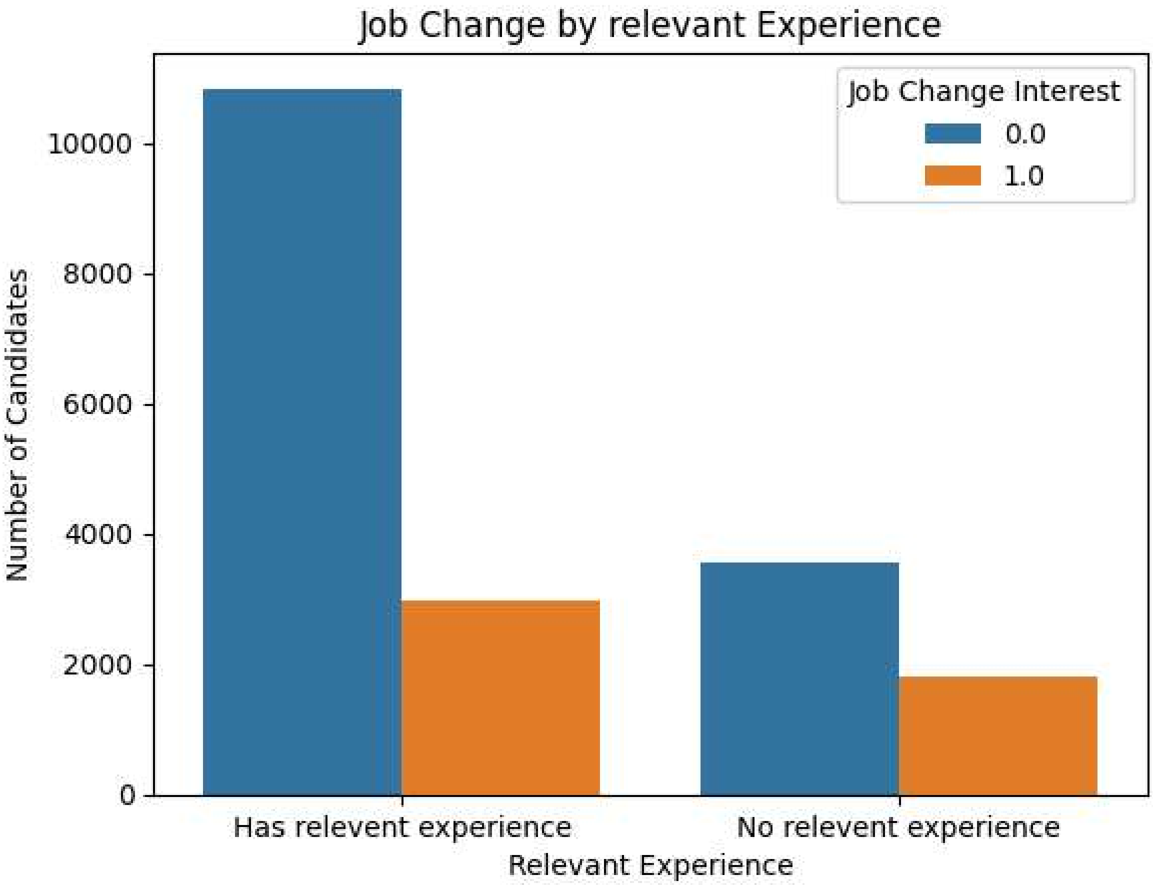
In [14]:

```
import seaborn as sns
import matplotlib.pyplot as plt

sns.countplot(x='target',data=df)
plt.title('Job Change Interest')
plt.show()
```

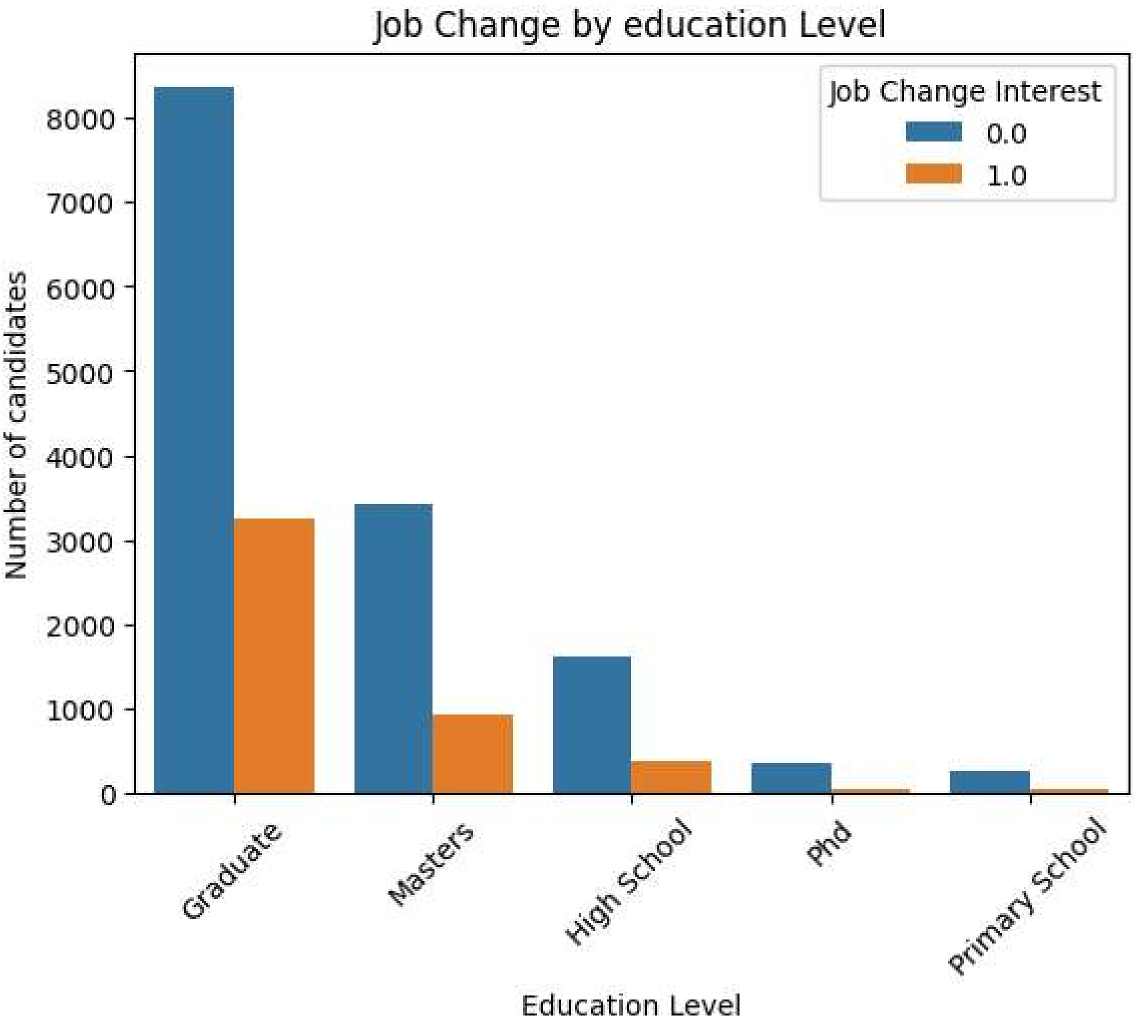


In [15]:
```
sns.countplot(x='relevent_experience',hue='target',data=df)
plt.title('Job Change by relevant Experience')
plt.xlabel('Relevant Experience')
plt.ylabel('Number of Candidates')
plt.legend(title='Job Change Interest')
plt.show()
```

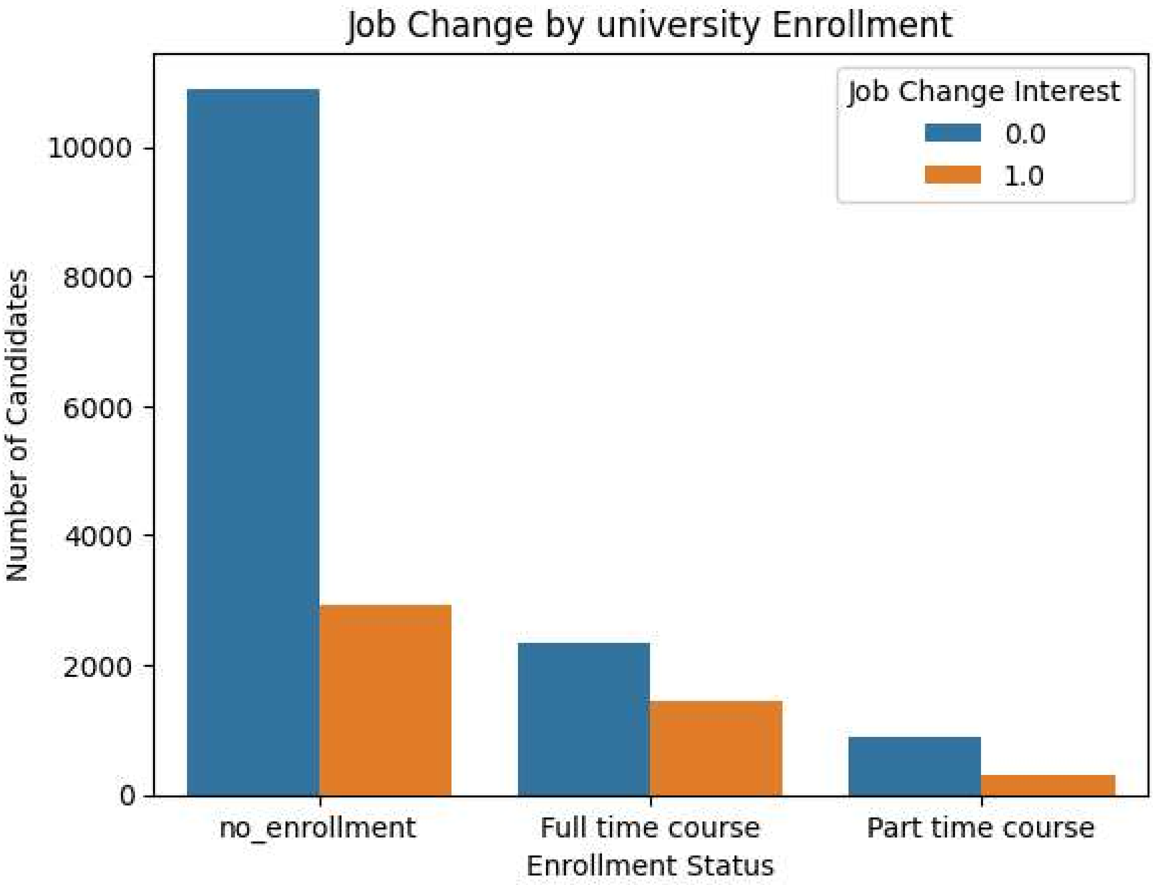


In [16]:

```
sns.countplot(x='education_level',hue='target',data=df)
plt.title('Job Change by education Level')
plt.xlabel('Education Level')
plt.ylabel('Number of candidates')
plt.xticks(rotation=45)
plt.legend(title='Job Change Interest')
plt.show()
```



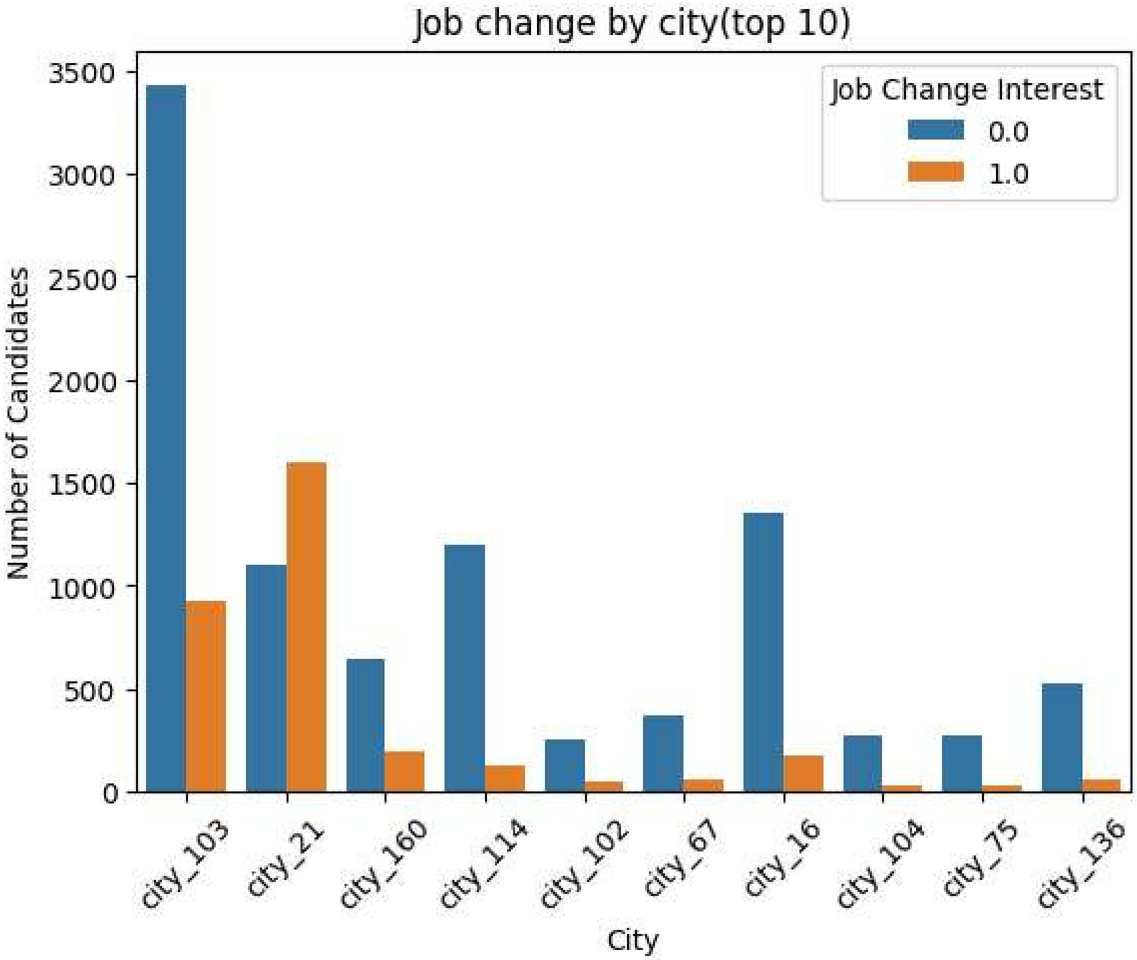
```
In [17]: sns.countplot(x='enrolled_university',hue='target',data=df)
         plt.title('Job Change by university Enrollment')
         plt.xlabel('Enrollment Status')
         plt.ylabel('Number of Candidates')
         plt.legend(title='Job Change Interest')
         plt.show()
```



In [18]:

```
top_cities = df['city'].value_counts().nlargest(10).index
df_city = df[df['city'].isin(top_cities)]

sns.countplot(x='city',hue='target',data=df_city)
plt.title('Job change by city(top 10)')
plt.xlabel('City')
plt.ylabel('Number of Candidates')
plt.xticks(rotation=45)
plt.legend(title='Job Change Interest')
plt.show()
```


Job change by city(top 10)
Job Change Interest
0.0
1.0
Number of Candidates
3500
3000
2500
2000
1500
1000
500
0
city_103
city_21
city_160
city_114
city_102
city_67
city_16
city_104
city_75
city_136
City

```
In [ ]:
```