In [ ]:

```
#  DEFAULT CREDIT
```

In [3]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [4]:

```
fullFilepath = ("C:/Users/pc/Videos/python/default_credit.xls")
data=pd.read_excel(fullFilepath)
```

In [5]:

```
data.head()
```

Out[5]:

	ID	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_0	PAY_2	PAY_3	PAY_4	...	BILL_A
0	1	20000	2	2	1	24	2	2	-1	-1	...	
1	2	120000	2	2	2	26	-1	2	0	0	...	
2	3	90000	2	2	2	34	0	0	0	0	...	1
3	4	50000	2	2	1	37	0	0	0	0	...	2
4	5	50000	1	2	1	57	-1	0	-1	0	...	2

5 rows × 25 columns

In [8]:

```
data.describe()
```

Out[8]:

	ID	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	
count	30000.000000	30000.000000	30000.000000	30000.000000	30000.000000	30000.000000	300
mean	15000.500000	167484.322667	1.603733	1.853133	1.551867	35.485500	
std	8660.398374	129747.661567	0.489129	0.790349	0.521970	9.217904	
min	1.000000	10000.000000	1.000000	0.000000	0.000000	21.000000	
25%	7500.750000	50000.000000	1.000000	1.000000	1.000000	28.000000	
50%	15000.500000	140000.000000	2.000000	2.000000	2.000000	34.000000	
75%	22500.250000	240000.000000	2.000000	2.000000	2.000000	41.000000	
max	30000.000000	1000000.000000	2.000000	6.000000	3.000000	79.000000	

8 rows × 25 columns

In [9]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30000 entries, 0 to 29999
Data columns (total 25 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   ID                          30000 non-null  int64
 1   LIMIT_BAL                   30000 non-null  int64
 2   SEX                         30000 non-null  int64
 3   EDUCATION                   30000 non-null  int64
 4   MARRIAGE                    30000 non-null  int64
 5   AGE                         30000 non-null  int64
 6   PAY_0                       30000 non-null  int64
 7   PAY_2                       30000 non-null  int64
 8   PAY_3                       30000 non-null  int64
 9   PAY_4                       30000 non-null  int64
 10  PAY_5                       30000 non-null  int64
 11  PAY_6                       30000 non-null  int64
 12  BILL_AMT1                   30000 non-null  int64
 13  BILL_AMT2                   30000 non-null  int64
 14  BILL_AMT3                   30000 non-null  int64
 15  BILL_AMT4                   30000 non-null  int64
 16  BILL_AMT5                   30000 non-null  int64
 17  BILL_AMT6                   30000 non-null  int64
 18  PAY_AMT1                    30000 non-null  int64
 19  PAY_AMT2                    30000 non-null  int64
 20  PAY_AMT3                    30000 non-null  int64
 21  PAY_AMT4                    30000 non-null  int64
 22  PAY_AMT5                    30000 non-null  int64
 23  PAY_AMT6                    30000 non-null  int64
 24  default payment next month  30000 non-null  int64
dtypes: int64(25)
memory usage: 5.7 MB
```

In [10]:
```
#checking for null
data.isnull().sum()
```

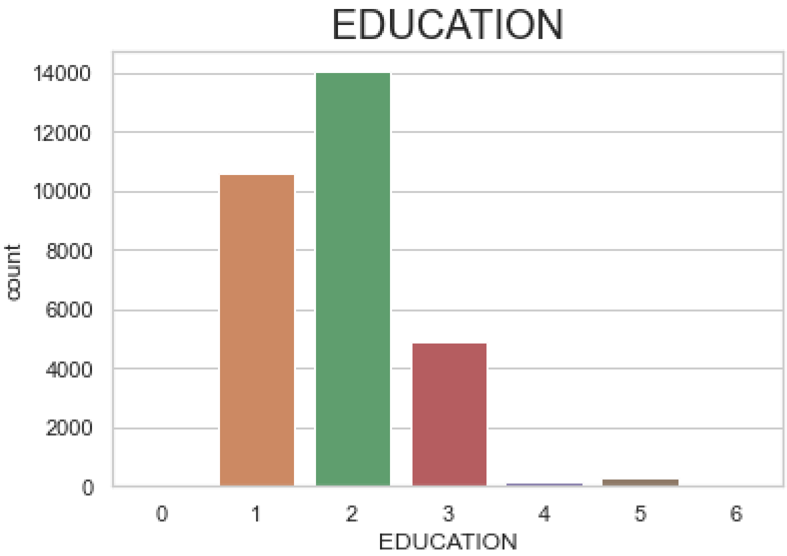
Out[10]:
```
ID                            0
LIMIT_BAL                     0
SEX                           0
EDUCATION                     0
MARRIAGE                      0
AGE                           0
PAY_0                         0
PAY_2                         0
PAY_3                         0
PAY_4                         0
PAY_5                         0
PAY_6                         0
BILL_AMT1                     0
BILL_AMT2                     0
BILL_AMT3                     0
BILL_AMT4                     0
BILL_AMT5                     0
BILL_AMT6                     0
PAY_AMT1                      0
PAY_AMT2                      0
PAY_AMT3                      0
PAY_AMT4                      0
PAY_AMT5                      0
PAY_AMT6                      0
default payment next month    0
dtype: int64
```

In [11]:
```
set(data.EDUCATION)
```

Out[11]:
```
{0, 1, 2, 3, 4, 5, 6}
```

```
In [42]: %matplotlib inline
         sns.countplot(data['EDUCATION'])
         plt.title(' EDUCATION ', fontsize = 20)
         plt.show()
```

```
C:\Users\pc\Downloads\anaconda\lib\site-packages\seaborn\_decorators.py:36: Fut
ureWarning: Pass the following variable as a keyword arg: x. From version 0.12,
the only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```



In [13]:
```
print(data['EDUCATION'].value_counts())
print()
print(data['EDUCATION'].value_counts(normalize=True)) #converts the results in pe
```

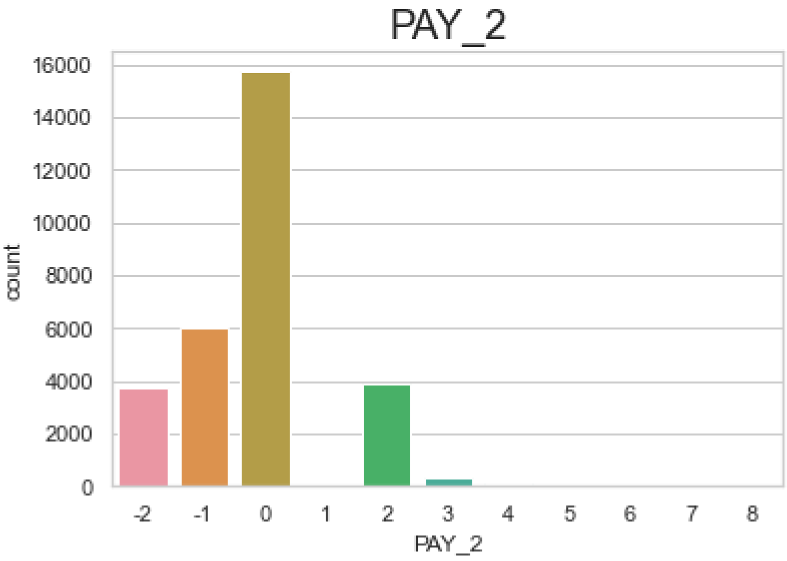
```
2    14030
1    10585
3     4917
5      280
4      123
6       51
0       14
Name: EDUCATION, dtype: int64

2    0.467667
1    0.352833
3    0.163900
5    0.009333
4    0.004100
6    0.001700
0    0.000467
Name: EDUCATION, dtype: float64
```

In [ ]:
```
the graph, it showed those that have higer level of education made their repaymen
```

In [43]:
```
%matplotlib inline
sns.countplot(data['PAY_2'])
plt.title('PAY_2', fontsize = 20)
plt.show()
```

```
C:\Users\pc\Downloads\anaconda\lib\site-packages\seaborn\_decorators.py:36: Fut
ureWarning: Pass the following variable as a keyword arg: x. From version 0.12,
the only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```



In [10]:

```
print(data['PAY_2'].value_counts())
print()
print(data['PAY_2'].value_counts(normalize=True))
```

```
 0    15730
-1     6050
 2     3927
-2     3782
 3      326
 4       99
 1       28
 5       25
 7       20
 6       12
 8        1
Name: PAY_2, dtype: int64

 0    0.524333
-1    0.201667
 2    0.130900
-2    0.126067
 3    0.010867
 4    0.003300
 1    0.000933
 5    0.000833
 7    0.000667
 6    0.000400
 8    0.000033
Name: PAY_2, dtype: float64
```

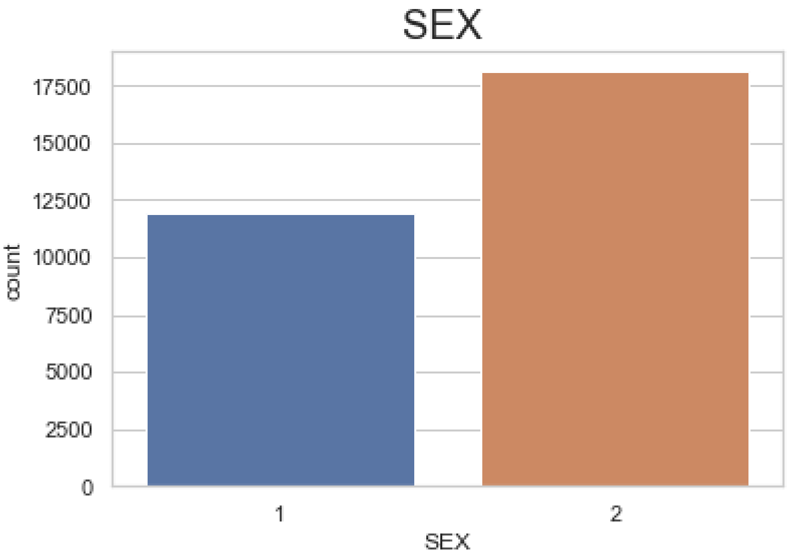
In [ ]:

```
# from the graph, it was shown that the repayment status for the month of August
```

In [44]:

```
%matplotlib inline
sns.countplot(data['SEX'])
plt.title('SEX ', fontsize = 20)
plt.show()
```

```
C:\Users\pc\Downloads\anaconda\lib\site-packages\seaborn\_decorators.py:36: Fut
ureWarning: Pass the following variable as a keyword arg: x. From version 0.12,
the only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```



In [49]:

```
#from the graph, it showed that "2" has a higher rate of repayments than "1"
```

In [7]:

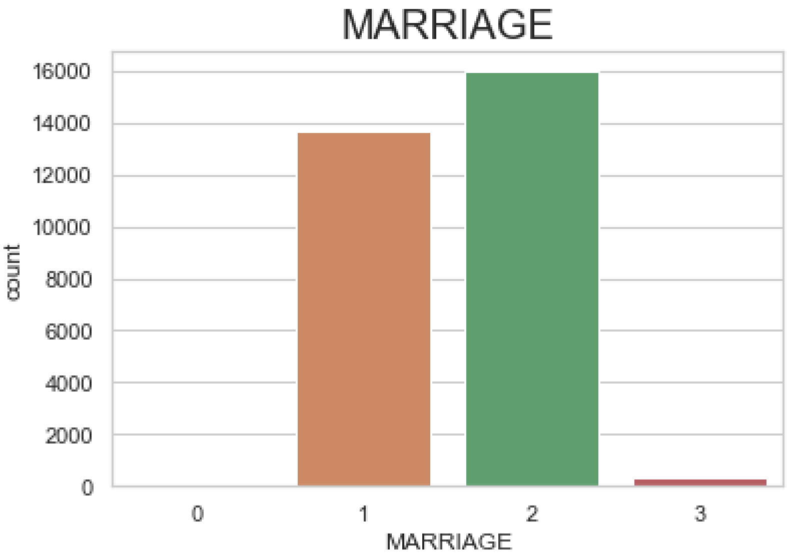
```
print(data['SEX'].value_counts())
print()
print(data['SEX'].value_counts(normalize=True))
```

```
2    18112
1    11888
Name: SEX, dtype: int64

2    0.603733
1    0.396267
Name: SEX, dtype: float64
```

In [45]:
```
%matplotlib inline
sns.countplot(data['MARRIAGE'])
plt.title('MARRIAGE', fontsize = 20)
plt.show()
```

```
C:\Users\pc\Downloads\anaconda\lib\site-packages\seaborn\_decorators.py:36: Fut
ureWarning: Pass the following variable as a keyword arg: x. From version 0.12,
the only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```



In [ ]:
```
#from the graph, the rate of defaulters is not based whether a person is married
```

In [10]:

```
print(data['MARRIAGE'].value_counts())
print()
print(data['MARRIAGE'].value_counts(normalize=True))
```

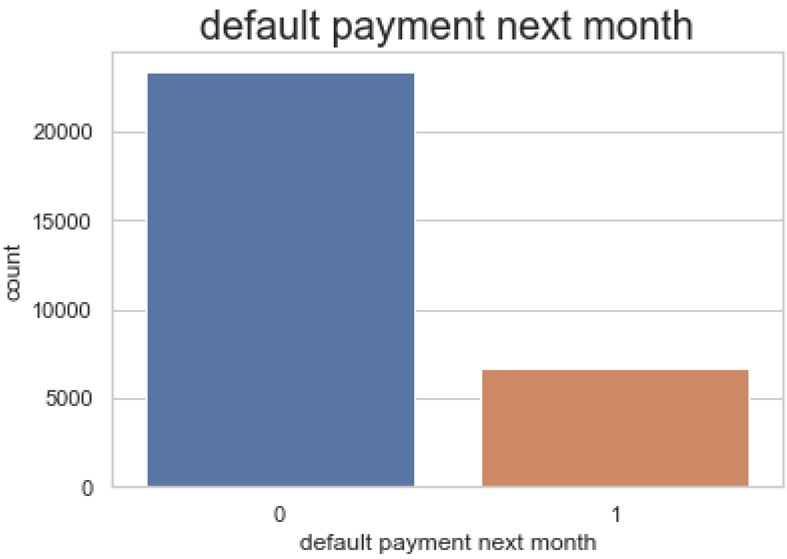
```
2    15964
1    13659
3      323
0       54
Name: MARRIAGE, dtype: int64

2    0.532133
1    0.455300
3    0.010767
0    0.001800
Name: MARRIAGE, dtype: float64
```

In [46]:

```
%matplotlib inline
sns.countplot(data['default payment next month'])
plt.title('default payment next month', fontsize = 20)
plt.show()
```

```
C:\Users\pc\Downloads\anaconda\lib\site-packages\seaborn\_decorators.py:36: Fut
ureWarning: Pass the following variable as a keyword arg: x. From version 0.12,
the only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```



In [12]:

```
print(data['default payment next month'].value_counts())
print()
print(data['default payment next month'].value_counts(normalize=True))
```

```
0    23364
1     6636
Name: default payment next month, dtype: int64

0    0.7788
1    0.2212
Name: default payment next month, dtype: float64
```

In [ ]:

```
#A total of 6,636 customers ended up not making repayments, while 23364 customers
This is calculated as follows:
6636/30000 * 100 = 22.12%
The overall number of visitors is 30000, Thus, the conversion rate is 22.12
```

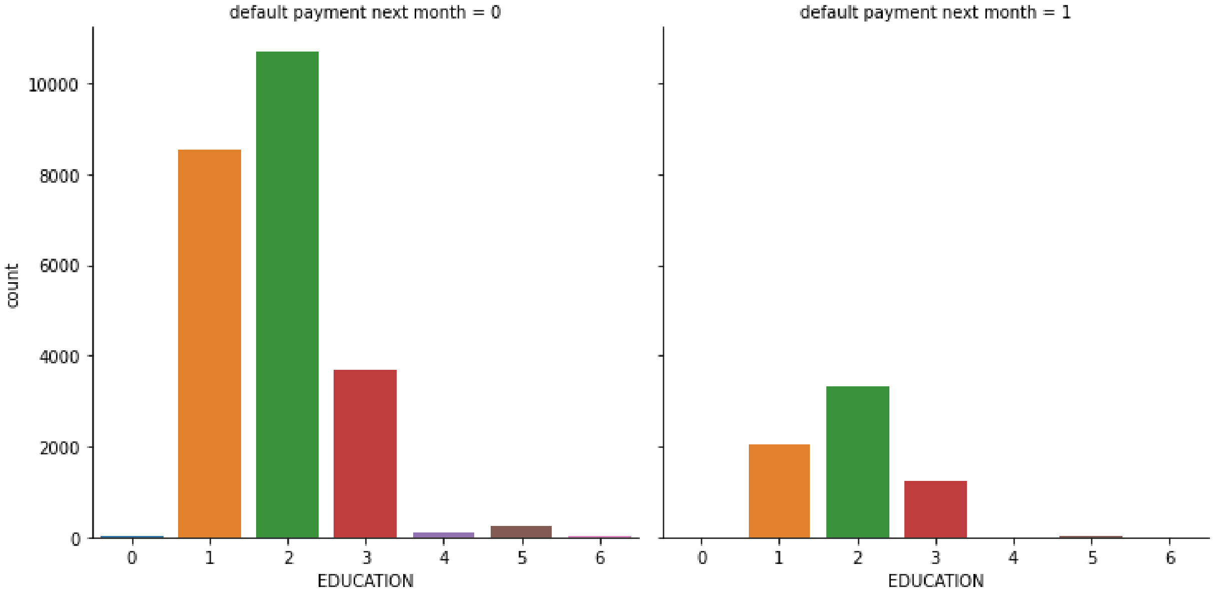
In [ ]:

```
#Bivariate Analysis
```

In [23]:

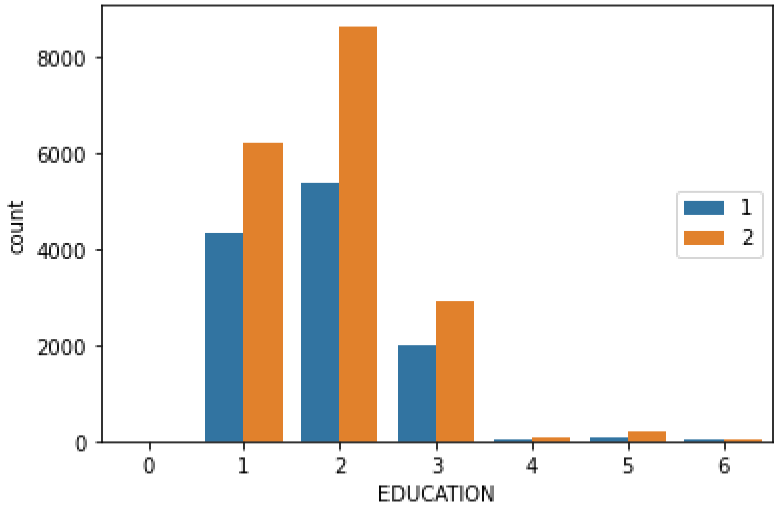
```
g = sns.catplot("EDUCATION", col="default payment next month", col_wrap=3, data=
plt.show()
```

```
C:\Users\pc\Downloads\anaconda\lib\site-packages\seaborn\_decorators.py:36: Fut
ureWarning: Pass the following variable as a keyword arg: x. From version 0.12,
the only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```



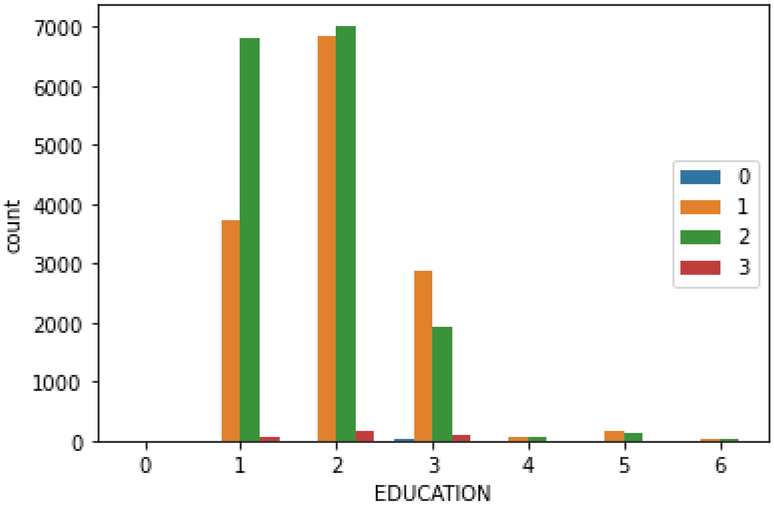
In [24]:

```
sns.countplot(x="EDUCATION", hue="SEX", data=data)
plt.legend(loc='right')
plt.show()
```



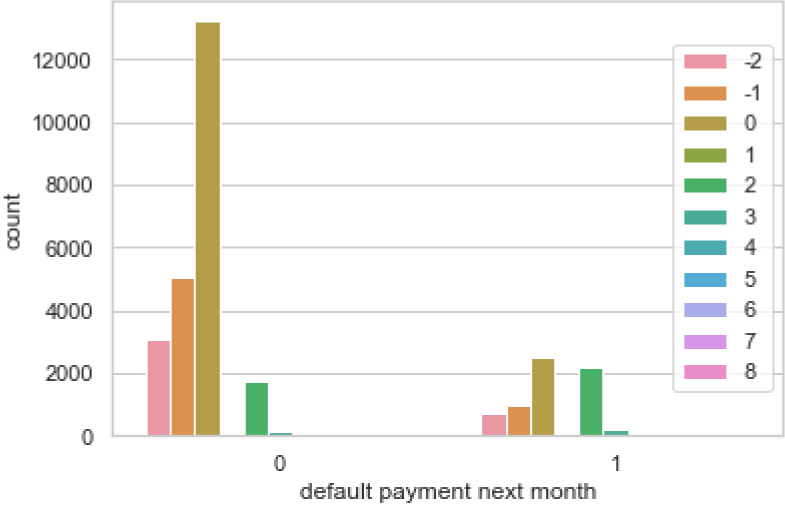
In [25]:

```
sns.countplot(x="EDUCATION", hue="MARRIAGE", data=data)
plt.legend(loc='right')
plt.show()
```



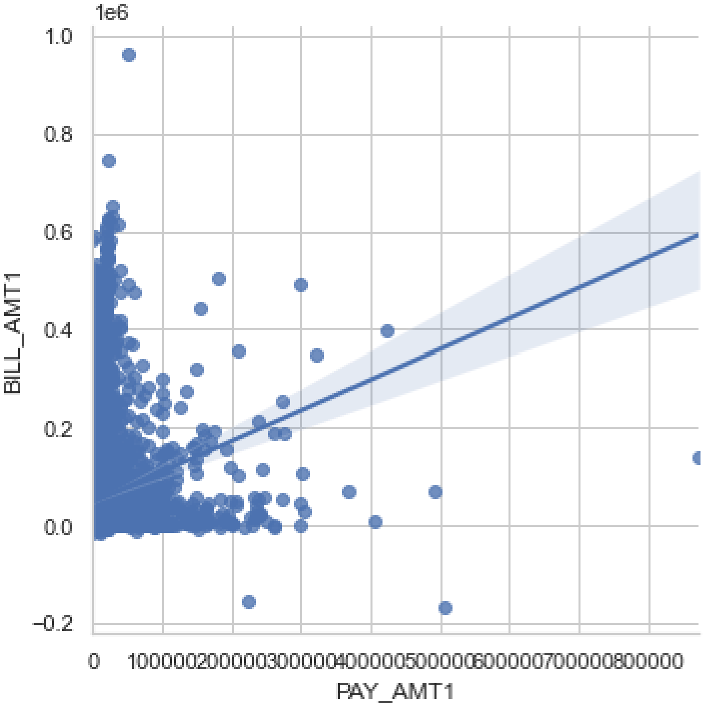
In [50]:

```
sns.countplot(x="default payment next month", hue="PAY_2", data=data)
plt.legend(loc='right')
plt.show()
```



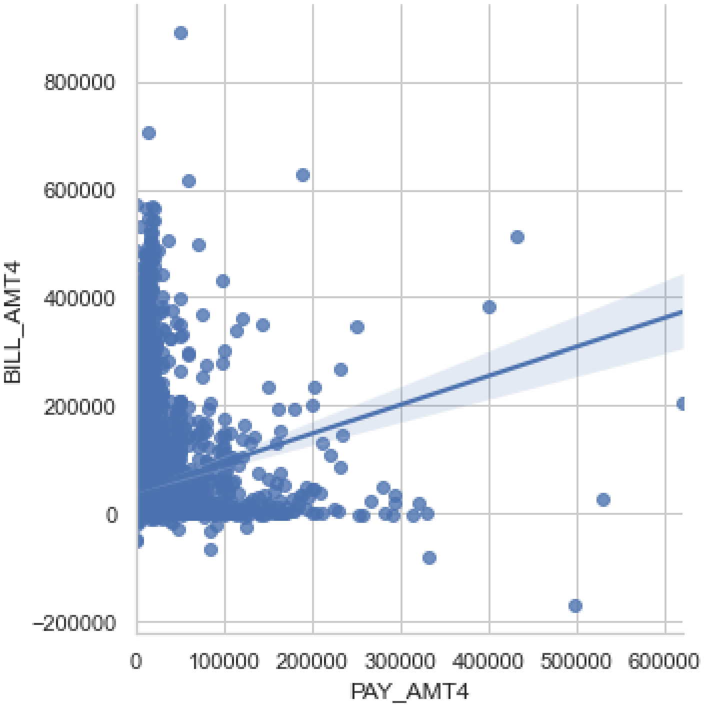
In [41]:
```
#CORRELATION

sns.set(style="whitegrid")
ax = sns.lmplot(x="PAY_AMT1", y="BILL_AMT1" , data=data)
```



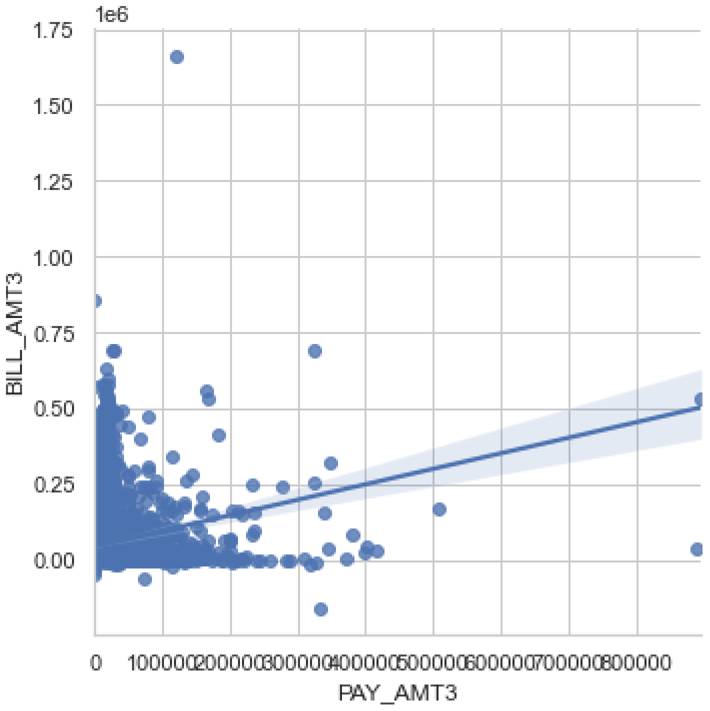
In [ ]:
```
#As you can see, there is a positive correlation between the BILL_AMT1 and the PA
```

In [40]:
```
sns.set(style="whitegrid")
ax = sns.lmplot(x="PAY_AMT4", y="BILL_AMT4" , data=data)
```



In [ ]:
```
a positive correlation between the BILL_AMT4 and the PAY_AMT4. With the increase
```

```
In [33]: sns.set(style="whitegrid")
         ax = sns.lmplot(x="PAY_AMT3", y="BILL_AMT3" , data=data)
```



```
In [ ]: #As you can see, there is a positive correlation between the BILL_AMT3 and the PA
```