Project 2: Titanic Dataset Analysis

Submitted By : Rahul Saxena

VARIABLE DESCRIPTIONS:

The dataset of study contains demographics and passenger information from 891 of the 2224 passengers and crew on board the Titanic. The variables included are:

survival	Survival (0 = No; 1 = Yes)
pclass	Passenger Class (1 = 1st; 2 = 2nd; 3 = 3rd)
name	Name
sex	Sex
age	Age
sibsp	Number of Siblings/Spouses Aboard
parch	Number of Parents/Children Aboard
ticket	Ticket Number
fare	Passenger Fare
cabin	Cabin
embarked	Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)

SPECIAL NOTES: Pclass is a proxy for socio-economic status (SES) 1st ~ Upper; 2nd ~ Middle; 3rd ~ Lower

Age is in Years; Fractional if Age less than One (1) If the Age is Estimated, it is in the form xx.5

With respect to the family relation variables (i.e. sibsp and parch) some relations were ignored. The following are the definitions used for sibsp and parch.

Sibling: Brother, Sister, Stepbrother, or Stepsister of Passenger Aboard Titanic Spouse: Husband or Wife of Passenger Aboard Titanic (Mistresses and Fiances Ignored) Parent: Mother or Father of Passenger Aboard Titanic Child: Son, Daughter, Stepson, or Stepdaughter of Passenger Aboard Titanic

Other family relatives excluded from this study include cousins, nephews/nieces, aunts/uncles, and in-laws. Some children travelled only with a nanny, therefore parch=0 for them. As well, some travelled with very close friends or neighbors in a village, however, the definitions do not support such relations.

Data Analysis :

Questions :

In this analysis we will try to answer some questions related to Survival rate according to :

1. Fare category
2. A person being Male or Female
3. Age of the person i.e, Child , Adult , Senior Citizen
4. Male Child or Female Child
4. Socio-economic status Upper Class (1st), Middle Class(2nd) , Lower Class(3rd)
5. Comparision of survival with respect to embarkment station
6. Chances of survival of Man with child(Father) or spouse(Husband) or Single?
7. Age-group of people with higher probablity of survival

Investigating Data

In [18]:
```
import pandas as pd
import numpy as np
titanic_data = pd.read_csv('titanic_data.csv')
```

In [19]:
```
titanic_data.head()
```

Out[19]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.25
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.92
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.05

Q1. Analysis of Survival on the basis of Ship Fare & Sex of a person

In [20]:
```
#No. of different fares in Titanic Ship
#Fare Variation

fare_list = titanic_data.Fare.unique()
fare_list = pd.DataFrame(fare_list)
#fare_list.describe()

no_survived = titanic_data['Survived'].value_counts()[1] #no of people survive
d

print no_survived
no_died = len(titanic_data)  - no_survived  #no of people died
print no_died
print('No of people Survived : {} , {:.2f}% of total'.format(no_survived, floa
t(no_survived*100 )/len(titanic_data)))
print('No of people Died : {} , {:.2f}% of total'.format(no_died, float(no_die
d*100 )/len(titanic_data)))
```

```
342
549
No of people Survived : 342 , 38.38% of total
No of people Died : 549 , 61.62% of total
```

In [21]:
```
fare_list.sort_values([0],inplace =True)
fares = pd.DataFrame(titanic_data.Fare)
fares.sort_values(['Fare'],inplace =True)

top_90_fare = fares[800:801]['Fare']  #Top 10% fare
top_90_fare
```

Out[21]:
```
102    77.2875
Name: Fare, dtype: float64
```

In [22]:
```
def isVIP(x):
    if x == 0:
        return "LowerClass" #Probably a Staff's relative/friend travelling wit
h passes
    elif x >= 77.2875:
        return "VIP" # One of Top 10% guys travelling in Ship
    else:
        return "Gen" #Normal People travelling in Ship

titanic_data["Is_VIP"] = pd.Series(titanic_data["Fare"].apply(isVIP), index=ti
tanic_data.index)
```

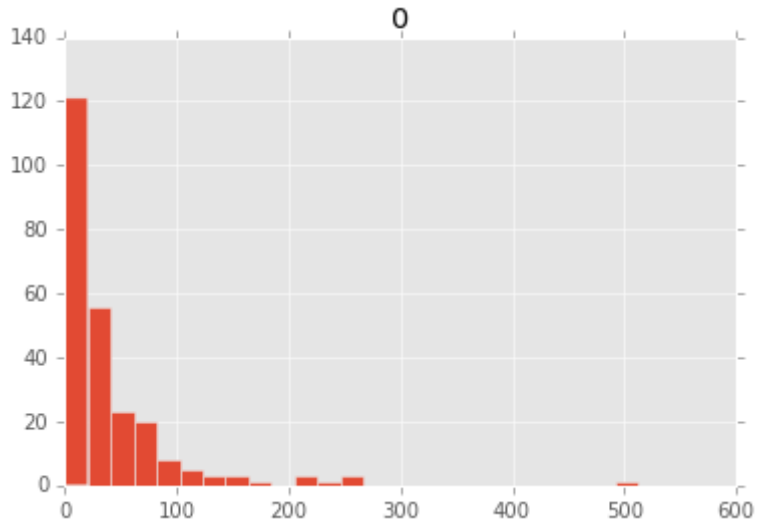
In [23]:
```
no_Gen = titanic_data['Is_VIP'].value_counts()['Gen']
no_Lower = titanic_data['Is_VIP'].value_counts()['LowerClass']
no_VIP = titanic_data['Is_VIP'].value_counts()['VIP']


no_Gen_survived =  titanic_data.groupby(['Is_VIP' , 'Survived']).size()['Gen']
[1]
no_Gen_died = no_Gen - no_Gen_survived

no_Lower_survived =  titanic_data.groupby(['Is_VIP' , 'Survived']).size()['Low
erClass'][1]
no_Lower_died = no_Lower - no_Lower_survived

no_VIP_survived =  titanic_data.groupby(['Is_VIP' , 'Survived']).size()['VIP']
[1]
no_VIP_died = no_VIP - no_VIP_survived

print('No. of General People with $0< fare < 77.28 : {}  , {:.2f}% of total'.f
ormat(no_Gen, float(no_Gen*100 )/len(titanic_data)))
print('No. of General People Survived : {} , {:.2f}%'.format(no_Gen_survived ,
 float(no_Gen_survived)*100/no_Gen))
print('No. of General People Died : {}, {:.2f}% '.format( no_Gen_died, float(n
o_Gen_died)*100/no_Gen))

print '\n'

print('No. of Lower Class People / Employees who were travelling for free : {}
  , {:.2f}% of total'.format(no_Lower, float(no_Lower*100
)/len(titanic_data)))
print('No. of Lower Class People/ Employees Survived : {} , {:.2f}%'.format(no
_Lower_survived , float(no_Lower_survived)*100/no_Lower))
print('No. of Lower Class People/ Employees Died : {}, {:.2f}% '.format( no_Lo
wer_died, float(no_Lower_died)*100/no_Lower))

print '\n'

print('No. of VIPs who were travelling : {}  , {:.2f}% of
total'.format(no_VIP, float(no_VIP*100 )/len(titanic_data)))
print('No. of VIPs Survived : {} , {:.2f}%'.format(no_VIP_survived , float(no_
VIP_survived)*100/no_VIP))
print('No. of VIPs Died : {}, {:.2f}% '.format( no_VIP_died,
float(no_VIP_died)*100/no_VIP))
```

```
No. of General People with $0< fare < 77.28 : 784  , 87.99% of total
No. of General People Survived : 272 , 34.69%
No. of General People Died : 512, 65.31%


No. of Lower Class People / Employees who were travelling for free : 15  , 1.
68% of total
No. of Lower Class People/ Employees Survived : 1 , 6.67%
No. of Lower Class People/ Employees Died : 14, 93.33%


No. of VIPs who were travelling : 92  , 10.33% of total
No. of VIPs Survived : 69 , 75.00%
No. of VIPs Died : 23, 25.00%
```

In [24]:
```
%pylab inline

import matplotlib
matplotlib.style.use('ggplot')
#fare_list.plot(kind="hist")
fare_list.hist([0],bins = 25)
```

```
Populating the interactive namespace from numpy and matplotlib
```

Out[24]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x0000000008110D30
>]], dtype=object)
```



The above analysis shows that if a person is travelling with *high price ticket* i.e, **VIPs** then their survival rate is much higher then **General & Lower Class / Employees**.

Q2. Comparision of Survival for Male & Female ?

In [25]:
```
#No. of males
no_male = titanic_data['Sex'].value_counts()['male']

no_female = titanic_data['Sex'].value_counts()['female']

#Survived/Died Male guys who survived
no_male_survived =  titanic_data.groupby(['Sex' , 'Survived']).size()[3]
no_male_died = no_male - no_male_survived

#Survived/Died Female guys who survived
no_female_survived =  titanic_data.groupby(['Sex' , 'Survived']).size()[1]
no_female_died = no_female - no_female_survived

print('No. of Males : {}  , {:.2f}% of total'.format(no_male, float(no_male*10
0 )/len(titanic_data)))
print('No. of Male Survived : {} , {:.2f}%'.format(no_male_survived , float(no
_male_survived)*100/no_male))
print('No. of Male Died : {}, {:.2f}% '.format( no_male_died, float(no_male_di
ed)*100/no_male ))

print '\n'
print('No. of Females : {}  , {:.2f}% of total'.format(no_female, float(no_fem
ale)*100 /len(titanic_data)))
print('No. of Female Survived : {} , {:.2f}%'.format(no_female_survived , floa
t(no_female_survived)*100/no_female))
print('No. of Female Died : {}, {:.2f}% '.format( no_female_died, float(no_fem
ale_died)*100/no_female ))
```

```
No. of Males : 577  , 64.76% of total
No. of Male Survived : 109 , 18.89%
No. of Male Died : 468, 81.11%


No. of Females : 314  , 35.24% of total
No. of Female Survived : 233 , 74.20%
No. of Female Died : 81, 25.80%
```

The above analysis gives an insight that Females were preferred to be saved i.e, their survival rate was high

Q3. Analysis of Survival according to age , i.e, Children | Adults | Senior Citizens

In [26]:
```
age_median = titanic_data.Age.median()
age_median

clean_data_age = titanic_data
clean_data_age.Age.fillna(age_median,inplace=True)

def isAge(x):
    if x < 18.0:
        return "Child"
    elif x >60.0:
        return "Senior Citizen"
    else:
        return "Adult"

clean_data_age["IsChild"] = pd.Series(clean_data_age["Age"].apply(isAge), inde
x=titanic_data.index)
titanic_data = clean_data_age
```

In [27]:
```
no_Child = titanic_data['IsChild'].value_counts()['Child']
no_SrCz = titanic_data['IsChild'].value_counts()['Senior Citizen']
no_Adult = titanic_data['IsChild'].value_counts()['Adult']

no_Child_survived =  titanic_data.groupby(['IsChild' , 'Survived']).size()['Ch
ild'][1]
no_Child_died = no_Child - no_Child_survived

#Survived/Died Female guys who survived
no_SrCz_survived =  titanic_data.groupby(['IsChild' , 'Survived']).size()['Sen
ior Citizen'][1]
no_SrCz_died = no_SrCz - no_SrCz_survived

no_Adult_survived =  titanic_data.groupby(['IsChild' , 'Survived']).size()['Ad
ult'][1]
no_Adult_died = no_Adult - no_Adult_survived


print('No. of Children : {}  , {:.2f}% of total'.format(no_Child, float(no_Chi
ld*100 )/len(titanic_data)))
print('No. of Child Survived : {} , {:.2f}%'.format(no_Child_survived ,
float(no_Child_survived)*100/no_Child))
print('No. of Child Died : {}, {:.2f}% '.format( no_Child_died, float(no_Child
_died)*100/no_Child ))

print '\n'

print('No. of Senior Citizen : {}  , {:.2f}% of total'.format(no_SrCz, float(n
o_SrCz*100 )/len(titanic_data)))
print('No. of Senior Citizen Survived : {} , {:.2f}%'.format(no_SrCz_survived
, float(no_SrCz_survived)*100/no_SrCz))
print('No. of Senior Citizen Died : {}, {:.2f}% '.format( no_SrCz_died,
float(no_SrCz_died)*100/no_SrCz ))

print '\n'

print('No. of Adults : {}  , {:.2f}% of total'.format(no_Adult,
float(no_Adult*100 )/len(titanic_data)))
print('No. of Adults Survived : {} , {:.2f}%'.format(no_Adult_survived ,
float(no_Adult_survived)*100/no_Adult))
print('No. of Adults  Died : {}, {:.2f}% '.format( no_Adult_died, float(no_Adu
lt_died)*100/no_Adult ))
```

```
No. of Children : 113  , 12.68% of total
No. of Child Survived : 61 , 53.98%
No. of Child Died : 52, 46.02%


No. of Senior Citizen : 22  , 2.47% of total
No. of Senior Citizen Survived : 5 , 22.73%
No. of Senior Citizen Died : 17, 77.27%


No. of Adults : 756  , 84.85% of total
No. of Adults Survived : 276 , 36.51%
No. of Adults  Died : 480, 63.49%
```

The Above Analysis shows that

- **54% of Children** were saved. So, survival of children were **higher than Adults and Senior Citizens.**
- Survival Rate of **Adults(36.5%)** is **higher** than **Senior Citizens(22.7%)**

Q4. Analysis of Survival in Male Child and Female Child

In [28]:
```
no_Female_Child = titanic_data.groupby(['IsChild' , 'Sex']).size()['Child']['female']
no_male_Child =  titanic_data.groupby(['IsChild' , 'Sex']).size()['Child']['male']

no_CFemale_Survived = titanic_data.groupby(['IsChild' ,'Survived', 'Sex']).size()['Child'][1]['female']
no_CFemale_Died = no_Female_Child - no_CFemale_Survived
no_CMale_Survived = titanic_data.groupby(['IsChild' ,'Survived',
'Sex']).size()['Child'][1]['male']
no_CMale_Died = no_male_Child -no_CMale_Survived


print('No. of Female Child : {}  , {:.2f}% of total'.format(no_Female_Child, float(no_Female_Child*100 )/len(titanic_data)))
print('No. of Female Child Survived : {} , {:.2f}%'.format(no_CFemale_Survived , float(no_CFemale_Survived)*100/no_Female_Child))
print('No. of Female Child Died : {}, {:.2f}% '.format( no_CFemale_Died, float(no_CFemale_Died)*100/no_Female_Child ))

print '\n'

print('No. of Male Child : {}  , {:.2f}% of total'.format(no_male_Child, float(no_male_Child*100 )/len(titanic_data)))
print('No. of Male Child Survived : {} , {:.2f}%'.format(no_CMale_Survived , float(no_CMale_Survived)*100/no_male_Child))
print('No. of Male Child Died : {}, {:.2f}% '.format( no_CMale_Died, float(no_CMale_Died)*100/no_male_Child ))

print '\n'
```

```
No. of Female Child : 55  , 6.17% of total
No. of Female Child Survived : 38 , 69.09%
No. of Female Child Died : 17, 30.91%


No. of Male Child : 58  , 6.51% of total
No. of Male Child Survived : 23 , 39.66%
No. of Male Child Died : 35, 60.34%
```

It shows that Survival Rate of **Female Children(69%) is more than Male Child(40%)**

Q5. Analysis of Survival according to Socio-economic status Upper Class (1st), Middle Class(2nd) , Lower Class(3rd)

In [29]:
```
no_class_1 = titanic_data['Pclass'].value_counts()[1]
no_class_2 = titanic_data['Pclass'].value_counts()[2]
no_class_3 = titanic_data['Pclass'].value_counts()[3]

#print titanic_data.groupby(['Pclass' , 'Survived']).size()

no_class_1_survived =  titanic_data.groupby(['Pclass' , 'Survived']).size()[1]
[1]
no_class_1_died = no_class_1 - no_class_1_survived

no_class_2_survived =  titanic_data.groupby(['Pclass' , 'Survived']).size()[2]
[1]
no_class_2_died = no_class_2 - no_class_2_survived

no_class_3_survived =  titanic_data.groupby(['Pclass' , 'Survived']).size()[3]
[1]
no_class_3_died = no_class_3 - no_class_3_survived

print('No. of Class 1 people : {}  , {:.2f}% of total'.format(no_class_1, floa
t(no_class_1*100 )/len(titanic_data)))
print('No. of Class 1 people Survived : {} , {:.2f}%'.format(no_class_1_surviv
ed , float(no_class_1_survived)*100/no_class_1))
print('No. of Class 1 people Died : {}, {:.2f}% '.format( no_class_1_died, flo
at(no_class_1_died)*100/no_class_1 ))

print '\n'
print('No. of Class 2 people : {}  , {:.2f}% of total'.format(no_class_2, floa
t(no_class_2)*100 /len(titanic_data)))
print('No. of Class 2 people Survived : {} , {:.2f}%'.format(no_class_2_surviv
ed , float(no_class_2_survived)*100/no_class_2))
print('No. of Class 2 people Died : {}, {:.2f}% '.format( no_class_2_died, flo
at(no_class_2_died)*100/no_class_2 ))

print '\n'
print('No. of Class 3 people : {}  , {:.2f}% of total'.format(no_class_3, floa
t(no_class_3)*100 /len(titanic_data)))
print('No. of Class 3 people Survived : {} , {:.2f}%'.format(no_class_3_surviv
ed , float(no_class_3_survived)*100/no_class_3))
print('No. of Class 3 people Died : {}, {:.2f}% '.format( no_class_3_died, flo
at(no_class_3_died)*100/no_class_3 ))
```

```
No. of Class 1 people : 216  , 24.24% of total
No. of Class 1 people Survived : 136 , 62.96%
No. of Class 1 people Died : 80, 37.04%


No. of Class 2 people : 184  , 20.65% of total
No. of Class 2 people Survived : 87 , 47.28%
No. of Class 2 people Died : 97, 52.72%


No. of Class 3 people : 491  , 55.11% of total
No. of Class 3 people Survived : 119 , 24.24%
No. of Class 3 people Died : 372, 75.76%
```

It shows that **Upper Class (63%)** were **preffered** over **Middle Class(47%) & Lower Class(24%)** people.

Q6. Chances of Survival According to Embarkment Station.

In [30]:
```
no_boarded_C = titanic_data['Embarked'].value_counts()['C']
no_boarded_Q = titanic_data['Embarked'].value_counts()['Q']
no_boarded_S = titanic_data['Embarked'].value_counts()['S']

#print titanic_data.groupby(['Embarked' , 'Survived']).size()

no_boarded_C_survived =  titanic_data.groupby(['Embarked' ,
'Survived']).size()['C'][1]
no_boarded_C_died = no_boarded_C - no_boarded_C_survived

no_boarded_Q_survived =  titanic_data.groupby(['Embarked' ,
'Survived']).size()['Q'][1]
no_boarded_Q_died = no_boarded_Q - no_boarded_Q_survived

no_boarded_S_survived =  titanic_data.groupby(['Embarked' ,
'Survived']).size()['S'][1]
no_boarded_S_died = no_boarded_S - no_boarded_S_survived

print('No. of People boarded from Cherbourg : {}  , {:.2f}% of total'.format(n
o_boarded_C, float(no_boarded_C*100 )/len(titanic_data)))
print('No. of People boarded from Cherbourg  who Survived: {} , {:.2f}%'.forma
t(no_boarded_C_survived , float(no_boarded_C_survived)*100/no_boarded_C))
print('No. of People boarded from Cherbourg  who Died : {}, {:.2f}% '.format(
no_boarded_C_died, float(no_boarded_C_died)*100/no_boarded_C ))

print '\n'
print('No. of People boarded from Queenstown : {}  , {:.2f}% of
total'.format(no_boarded_Q, float(no_class_2)*100 /len(titanic_data)))
print('No. of People boarded from Queenstown  who Survived : {} , {:.2f}%'.for
mat(no_boarded_Q_survived , float(no_boarded_Q_survived)*100/no_boarded_Q))
print('No. of People boarded from Queenstown  who Died : {}, {:.2f}% '.format(
 no_boarded_Q_died, float(no_boarded_Q_died)*100/no_boarded_Q ))

print '\n'
print('No. of People boarded from Southampton : {}  , {:.2f}% of
total'.format(no_boarded_S, float(no_boarded_S)*100 /len(titanic_data)))
print('No. of People boarded from Southampton  who Survived : {} , {:.2f}%'.fo
rmat(no_boarded_S_survived , float(no_boarded_S_survived)*100/no_boarded_S))
print('No. of People boarded from Southampton  who Died : {}, {:.2f}%
'.format( no_boarded_S_died, float(no_boarded_S_died)*100/no_boarded_S ))
```

```
No. of People boarded from Cherbourg : 168  , 18.86% of total
No. of People boarded from Cherbourg  who Survived: 93 , 55.36%
No. of People boarded from Cherbourg  who Died : 75, 44.64%


No. of People boarded from Queenstown : 77  , 20.65% of total
No. of People boarded from Queenstown  who Survived : 30 , 38.96%
No. of People boarded from Queenstown  who Died : 47, 61.04%


No. of People boarded from Southampton : 644  , 72.28% of total
No. of People boarded from Southampton  who Survived : 217 , 33.70%
No. of People boarded from Southampton  who Died : 427, 66.30%
```

It shows that people who boarded from :

- **Cherbourg** had highed probablity of survival**(55.36%)**
- **Southampton** had lowest probablity of survival**(33.7%)**

Q7. Chances of survival of Man with child(Father) or spouse(Husband) or Single

In [31]:

```
def isAdultMan(x):
    return (x["IsChild"] =="Senior Citizen" or x["IsChild"] =="Adult")  and
x["Sex"] == "male"

adult_man_titanic_data = titanic_data[titanic_data.apply(isAdultMan, axis=1)]


def isFamilyMan(x):
    if x["SibSp"] > 0:
        if x["Parch"] > 0:
            return "Father"
        else:
            return "Husband"
    else:
        return "Single"
adult_man_titanic_data["FamilyMan"] = pd.Series(adult_man_titanic_data.apply(i
sFamilyMan, axis=1), index=adult_man_titanic_data.index)
# print adult_man_titanic_data["FamilyMan"].value_counts()

no_Adult_Fathers_survived =  adult_man_titanic_data.groupby(['FamilyMan' , 'Su
rvived']).size()['Father'][1]
no_Adult_Fathers_died = adult_man_titanic_data.groupby(['FamilyMan' , 'Survive
d']).size()['Father'][0]

no_Fathers = adult_man_titanic_data["FamilyMan"].value_counts()["Father"]

print('No. of Adult Fathers : {}  , {:.2f}% of total'.format(no_Fathers , floa
t(no_Fathers *100 )/len(titanic_data)))
print('No. of Adult Fathers Survived : {} , {:.2f}%'.format(no_Adult_Fathers_s
urvived , float(no_Adult_Fathers_survived)*100/no_Fathers))
print('No. of Adult Fathers Died : {}, {:.2f}% '.format(
no_Adult_Fathers_died, float(no_Adult_Fathers_died)*100/no_Fathers ))

print '\n'

no_Husband_survived =  adult_man_titanic_data.groupby(['FamilyMan' , 'Survive
d']).size()['Husband'][1]
no_Husband_died = adult_man_titanic_data.groupby(['FamilyMan' , 'Survived']).s
ize()['Husband'][0]


no_Husband = adult_man_titanic_data["FamilyMan"].value_counts()["Husband"]

print('No. of Adult Husband : {}  , {:.2f}% of total'.format(no_Husband , floa
t(no_Husband *100 )/len(titanic_data)))
print('No. of Adult Husband Survived : {} , {:.2f}%'.format(no_Husband_survive
d , float(no_Husband_survived)*100/no_Husband))
print('No. of Adult Husband Died : {}, {:.2f}% '.format( no_Husband_died, floa
t(no_Husband_died)*100/no_Husband ))

print '\n'

no_Single_survived =  adult_man_titanic_data.groupby(['FamilyMan' ,
'Survived']).size()['Single'][1]
no_Single_died = adult_man_titanic_data.groupby(['FamilyMan' , 'Survived']).si
ze()['Single'][0]
```

```
no_Single = adult_man_titanic_data["FamilyMan"].value_counts()["Single"]

print('No. of Adult Single : {}  , {:.2f}% of total'.format(no_Single , 
float(no_Single *100 )/len(titanic_data)))
print('No. of Adult Single Survived : {} , {:.2f}%'.format(no_Single_survived
, float(no_Single_survived)*100/no_Single))
print('No. of Adult Single Died : {}, {:.2f}% '.format( no_Single_died, 
float(no_Single_died)*100/no_Single ))

print '\n'
```

```
No. of Adult Fathers : 34  , 3.82% of total
No. of Adult Fathers Survived : 6 , 17.65%
No. of Adult Fathers Died : 28, 82.35%


No. of Adult Husband : 70  , 7.86% of total
No. of Adult Husband Survived : 15 , 21.43%
No. of Adult Husband Died : 55, 78.57%


No. of Adult Single : 415  , 46.58% of total
No. of Adult Single Survived : 65 , 15.66%
No. of Adult Single Died : 350, 84.34%



c:\python27_x64\lib\site-packages\ipykernel\__main__.py:15: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/st
able/indexing.html#indexing-view-versus-copy
```

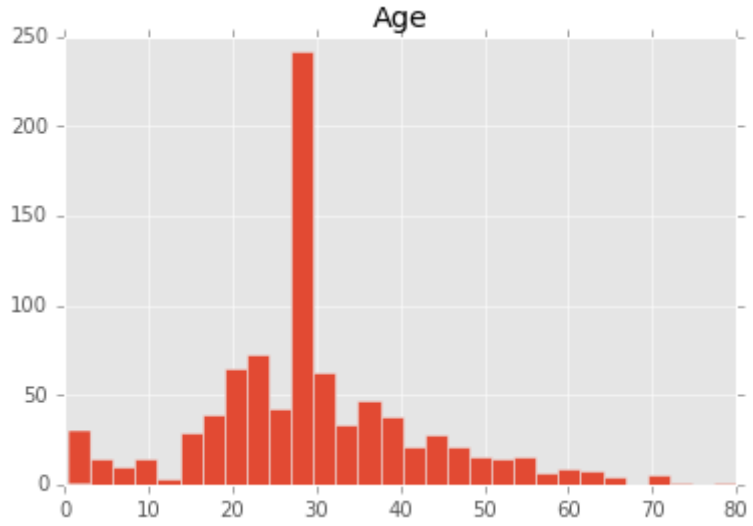
It shows that Suvival rate of Man travelling with their Wife is Higher that Man travelling with their kids

---

Q7. Best Age-group which had highest probablity of survival

In [32]:
```
Best_Age = clean_data_age[clean_data_age['Survived']==1]['Age']
Best_Age


age_list = titanic_data.Age
age_list = pd.DataFrame(age_list)
#fare_list.describe()

age_list.hist([0],bins = 30)
```

Out[32]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x0000000008B287B8
>]], dtype=object)
```



It shows that **(27 - 30) age group** people survived more