In [1]:
```
import pandas as pd
from matplotlib import pyplot as plt
import numpy as np
import math
```

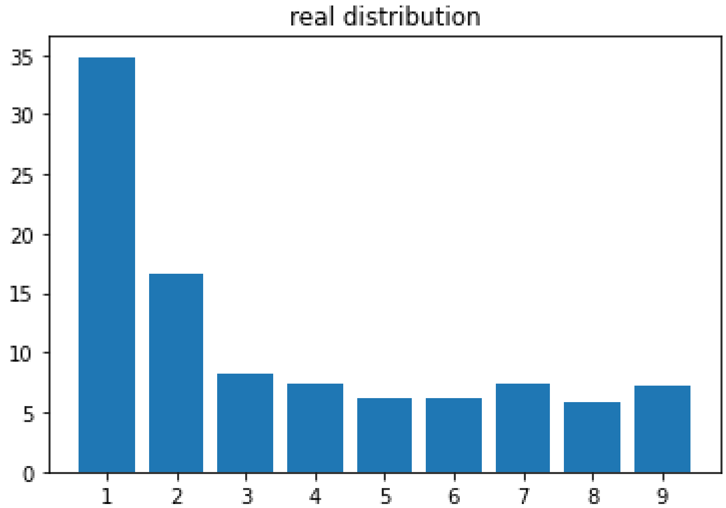
In [2]:
```
df=pd.read_csv("FashionDataset.csv")
#print(df)

df["MRP"]=df["MRP"].replace("Nan",0)

df["SellPrice"]=df["SellPrice"].replace("Nan",0)
```

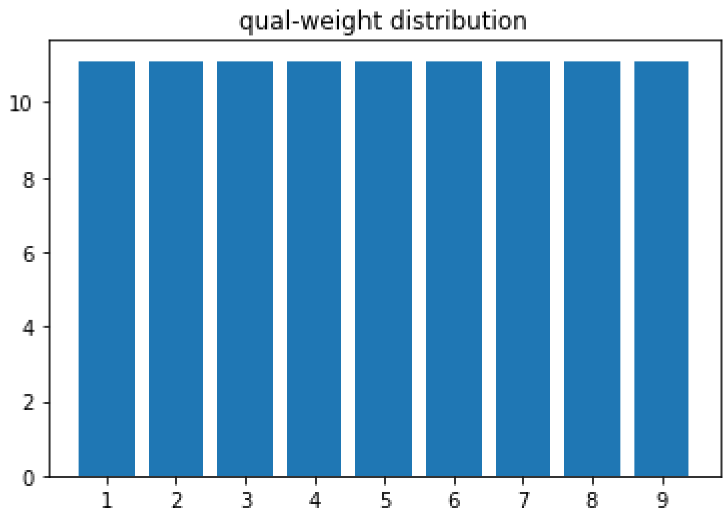
Question#1. plot 3 histograms for the frequencies for real distribution, equal-weight and Bernford (for each digit)

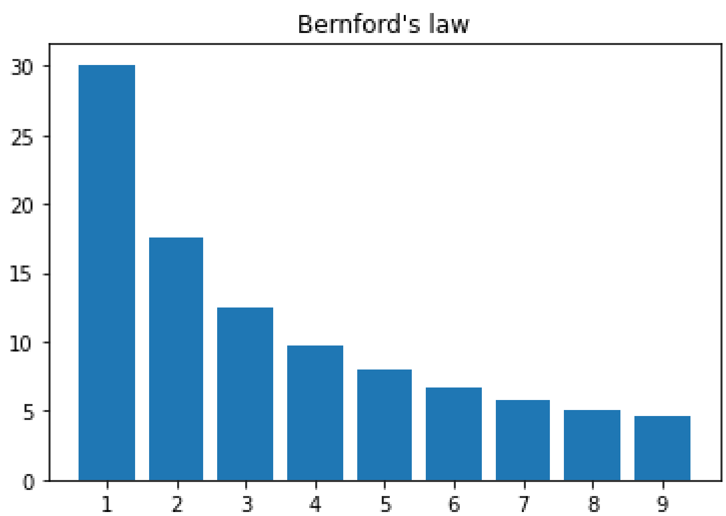
In [3]:
```
df_disp=df
```

In [4]:
```
dig_list=list(range(1,10))
dig_list_hist=[]
dig_count=[0,0,0,0,0,0,0,0,0]
bf_law=[ 30.1,17.6,12.5,9.7,7.9,6.7,5.8,5.1,4.6]
sell_price=df_disp["SellPrice"].values
print(sell_price)
for i in dig_list:
    count=0
    for j in sell_price:
        #print(str(j)[0])
        if i == int(str(j)[0]):
            dig_list_hist.append(int(str(j)[0]))
            count+=1
            dig_count[i-1]=count

#print(str(df_disp["MRP"][0])[0])
real_dis=[0,0,0,0,0,0,0,0,0]
dig_ew=[]
for i in range(len(dig_count)):
    real_dis[i]=(dig_count[i]/sum(dig_count))*100
    dig_ew.append(11.11)


plt.title("real distribution")
plt.xticks(dig_list)
plt.bar(x=dig_list,height=real_dis)
plt.show()

plt.bar(x=dig_list,height=dig_ew)
plt.xticks(dig_list)
plt.title("qual-weight distribution")
plt.show()

plt.title("Bernford's law")
plt.xticks(dig_list)
plt.bar(x=dig_list,height=bf_law)
plt.show()
```

```
['849' '2449' '599' ... '643' '2799' '1899']
```



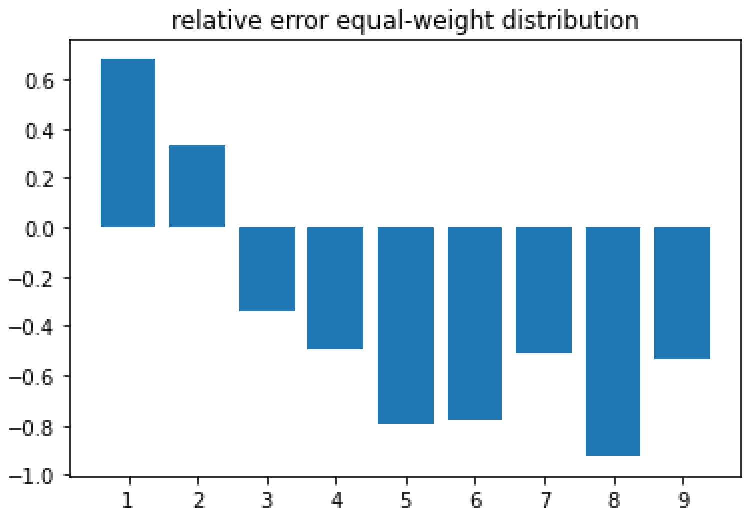




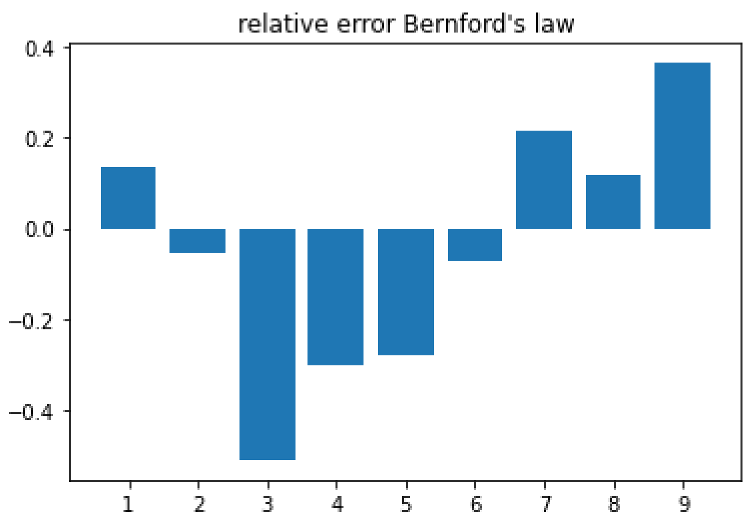
Question#2. plot 2 histograms for the relative errors for Models 1 and 2 (for each digit)

In [5]:
```
from codecs import ascii_decode
```

```
RE_rate_m1=[]
RE_rate_m2=[]

for i in range(len(dig_list)):
    RE_rate_m1.append((real_dis[i] - dig_ew[i])/real_dis[i])
    RE_rate_m2.append((real_dis[i] - bf_law[i])/real_dis[i])

plt.bar(x=dig_list,height=RE_rate_m1)
plt.xticks(dig_list)
plt.title("relative error equal-weight distribution")
plt.show()
print(RE_rate_m1)


plt.title("relative error Bernford's law")
plt.xticks(dig_list)
plt.bar(x=dig_list,height=RE_rate_m2)
plt.show()
```




```
[0.6809919902912621, 0.3329714778725132, -0.34277993461381273, -0.4948964968152864, -0.79845785440613, -0.7770592212006489, -0.5072396788990825, -0.9237602459016392, -0.5354123831775699]
```



Question#3compute RMSE (root mean squared error) for model 1 and 2. Which model is closer to the real distribution?

In [6]:
```
RMSE_m2=0
RMSE_m1=0
sum_m2=0
sum_m1=0
for i in range(len(RE_rate_m2)):
    sum_m2+=(RE_rate_m2[i]**2)
    sum_m1+=(RE_rate_m1[i]**2)
RMSE_m2=math.sqrt(sum_m2/len(RE_rate_m2))
RMSE_m1=math.sqrt(sum_m1/len(RE_rate_m1))
print("RMSE for model 1 is ",RMSE_m1)
print("RMSE for model 2 is ",RMSE_m2)
print("From my point of view, model 2(Bernford' law) is more close to the real
```

```
RMSE for model 1 is  0.6303163407179797
RMSE for model 2 is  0.26881150887719035
From my point of view, model 2(Bernford' law) is more close to the real distri
bution
```

Question#4 take 3 categories of your choice For each of these categories do the following: (a) compute F , P and π. (b) using RMSE as a ”distance” metric, for which of these chosen three coun– tries is the distribution ”closest” to equal weight P ?

In [7]:
```
inital=[]
for i in range(len(df)):
    inital.append(int(str(df["SellPrice"][i])[0]))
#print(inital)
df_disp["inital"]=inital
Category =df["Category"].unique()
total_sum={}
for i in Category:
    for j in dig_list:
        #test=df.loc[(df["Category"]==i)&(df["inital"]==j),:].sum()
        total_sum[(i,j)]=int(df.loc[(df["Category"]==i)&(df["inital"]==j),:]["i
#print(total_sum)
total_sum_res={}
for i in Category:
    count=0
    for j in dig_list:
        count+=total_sum[(i,j)]
        total_sum_res[i]=count
#print(total_sum_res)
Q4_real_dis={}
for i in Category:
    temp=[]
    for j in dig_list:
        a=(total_sum[((i,j))]/total_sum_res[i])*100
        temp.append(a)
        #print(temp)
        Q4_real_dis[i]=temp
print("for real distribution is",Q4_real_dis)
Q4_RE_rate_m1={}
for i in Category:
    temp=[]
    for j in range(len(dig_list)):
        a=((Q4_real_dis[i][j]-dig_ew[j])/Q4_real_dis[i][j])
```

```
        temp.append(a)
        Q4_RE_rate_m1[i]=temp
print("\n","for moled 1 RE is ",Q4_RE_rate_m1)

Q4_RE_rate_m2={}
for i in Category:
    temp=[]
    for j in range(len(dig_list)):
        a=((Q4_real_dis[i][j]-bf_law[j])/Q4_real_dis[i][j])
        temp.append(a)
        Q4_RE_rate_m2[i]=temp
print("\n","for moled 2 RE is ",Q4_RE_rate_m2)

Q4_squr={}
for i in Category:
    temp=[]
    for j in range(len(dig_list)):
        a=Q4_RE_rate_m1[i][j]**2
        temp.append(a)
        Q4_squr[i]=temp
print("\n","for moled 2 RE is ",Q4_squr)

Q4_final_res={}
for i in Category:
    temp=[]
    for j in range(len(dig_list)):
        a=math.sqrt(sum(Q4_squr[i])/9)
        Q4_final_res[i]=a
print("\n","Final RMSE for equal weight P is",Q4_final_res)
print("\n","three coun- tries is the distribution ”closest” to equal weight P i
```

```
for real distribution is {'Westernwear-Women': [40.64160401002506, 13.13283208
0200501, 6.025062656641604, 6.345864661654136, 4.4411027568922306, 5.654135338
345865, 7.639097744360902, 6.37593984962406, 9.74436090225564], 'Indianwear-Wo
men': [36.80200501253133, 18.897243107769423, 6.5964912280701755, 6.0852130325
814535, 5.694235588972431, 5.674185463659147, 7.9298245614035086, 6.1453634085
21303, 6.175438596491229], 'Lingerie&Nightwear-Women': [25.519379844961243, 1
6.74418604651163, 9.891472868217054, 9.178294573643411, 10.48062015503876, 8.1
24031007751938, 8.682170542635658, 5.550387596899225, 5.829457364341086], 'Foo
twear-Women': [31.11111111111111, 19.959595959595962, 11.95959595959596, 9.292
929292929292, 5.454545454545454, 6.707070707070708, 3.878787878787879, 4.24242
4242424243, 7.3939393939393945], 'Watches-Women': [27.362318840579707, 18.4347
82608695652, 10.898550724637682, 9.971014492753623, 8.17391304347826, 9.043478
260869566, 6.840579710144927, 4.521739130434783, 4.753623188405797], 'Fragranc
e-Women': [13.894736842105262, 8.421052631578947, 17.894736842105264, 15.78947
3684210526, 13.263157894736842, 12.842105263157894, 8.421052631578947, 5.89473
6842105263, 3.578947368421052], 'Jewellery-Women': [25.73913043478261, 19.6521
73913043477, 17.391304347826086, 10.72463768115942, 8.057971014492754, 4.28985
50724637685, 5.391304347826087, 4.0, 4.753623188405797]}

 for moled 1 RE is  {'Westernwear-Women': [0.7266348051307351, 0.1540286259541
985, -0.8439642262895173, -0.7507464454976301, -1.5016309255079006, -0.9649335
106382976, -0.45436023622047234, -0.7424882075471698, -0.1401466049382715], 'I
ndianwear-Women': [0.6981142740397712, 0.41208355437665783, -0.684228723404255
2, -0.8257372322899506, -0.9510959507042254, -0.9579902826855126, -0.401039823
00884955, -0.8078670473083198, -0.7990624999999998], 'Lingerie&Nightwear-Wome
n': [0.564644592952612 5, 0.3364861111111112, -0.12318965517241376, -0.21046452
702702692, -0.06005177514792885, -0.36754770992366403, -0.2796339285714286, -
1.0016620111731842, -0.9058377659574465], 'Footwear-Women': [0.642892857142857
2, 0.44337550607287457, 0.07103885135135145, -0.19553260869565217, -1.03683333
33333333, -0.6564608433734938, -1.8642968749999997, -1.6187857142857138, -0.50
25819672131145], 'Watches-Women': [0.5939671610169491, 0.397334905660377 4, -0.
019401595744680 76, -0.11422965116279066, -0.35920212765957454, -0.228509615384
61517, -0.6241313559322035, -1.4570192307692305, -1.3371646341463415], 'Fragra
nce-Women': [0.20041666666666663, -0.3193125, 0.37914705882352945, 0.296366666
66666667, 0.1623412698412699, 0.13487704918032786, -0.3193125, -0.884732142857
1429, -2.104264705882353], 'Jewellery-Women': [0.5683614864864865, 0.434668141
59292033, 0.361175, -0.03593243243243241, -0.37875899280575526, -1.58983108108
10807, -1.0607258064516127, -1.7774999999999999, -1.3371646341463415]}

 for moled 2 RE is  {'Westernwear-Women': [0.25937962506166745, -0.34015267175
572533, -1.0746672212978368, -0.5285545023696681, -0.7788374717832958, -0.1849
7340425531905, 0.24074803149606303, 0.2001179245283019, 0.5279320987654321],
'Indianwear-Women': [0.18210977935167533, 0.06864721485411131, -0.894946808510
6382, -0.5940280065897857, -0.38736795774647903, -0.18078621908127226, 0.26858
40707964602, 0.17010603588907014, 0.2551136363636365], 'Lingerie&Nightwear-Wom
en': [-0.17949574726609957, -0.05111111111111114, -0.263714733542319 8, -0.056
841216216216 1, 0.2462278106508876, 0.17528625954198473, 0.331964285714285 7, 0.
0811452513966481 2, 0.21090425531914908], 'Footwear-Women': [0.032499999999999
4, 0.1182186234817814 2, -0.04518581081081076, -0.04380434782608694, -0.4483333
333333353, 0.001054216867469936, -0.49531249999999993, -0.20214285714285687,
0.3778688524590164 7], 'Watches-Women': [-0.10005296610169508, 0.0452830188679 2
4456, -0.1469414893617021, 0.027180232558139592, 0.03351063829787218, 0.259134
61538461546, 0.15211864406779654, -0.12788461538461518, 0.03231707317073174],
'Fragrance-Women': [-1.1662878787878792, -1.0900000000000003, 0.30147058823529
416, 0.3856666666666667, 0.4043650793650793 7, 0.4782786885245901, 0.3112499999
9999997, 0.1348214285714286, -0.28529411764705886], 'Jewellery-Women': [-0.169
4256756756757, 0.10442477876106 18, 0.2812499999999999 4, 0.09554054054054058,
0.019604316546762576, -0.5618243243243243, -0.0758064516129032, -0.27499999999
99999, 0.03231707317073174]}
```

```
 for moled 2 RE is  {'Westernwear-Women': [0.5279981400273813, 0.0237248176133
38393, 0.7122756152564635, 0.5636202254273261, 2.254895436441714, 0.9310966799
527496, 0.20644322425832343, 0.5512887383466091, 0.019641070875723945], 'India
nwear-Women': [0.48736353961807677, 0.16981285578769992, 0.4681689459314168,
0.6818419767898678, 0.9045835074459743, 0.9177453817198683, 0.1608329396389693
9, 0.652649166126663, 0.6385008789062496], 'Lingerie&Nightwear-Women': [0.3188
2351635062145, 0.11322290297067908, 0.015175691141498209, 0.04429531713671014
4, 0.0036062156984174054, 0.1350913190701299, 0.07819513400829084, 1.003326784
6275082, 0.8205420582347775], 'Footwear-Women': [0.41331122576530616, 0.196581
83938537763, 0.005046518401319407, 0.038233001063327034, 1.075023361111111, 0.
4309408388826387, 3.4756028381347646, 2.6204671887755087, 0.2525886337678041
6], 'Watches-Women': [0.3527969883665344, 0.157875027256141, 0.000376421917440
0145, 0.013048413204772841, 0.1290261685151653, 0.05221664432322475, 0.3895399
494577709, 2.12290503883136, 1.7880092588117191], 'Fragrance-Women': [0.040166
84027777776, 0.10196047265625, 0.1437524922145329, 0.0878332011111111, 0.02635
4687893676006, 0.01819181839559258, 0.10196047265625, 0.7827509646045918, 4.42
7929952422146], 'Jewellery-Women': [0.3230347793211286, 0.18893639331584303,
0.13044738062500003, 0.0012911397005113206, 0.14345837463123015, 2.52756286637
1438, 1.1251392364724242, 3.1595062499999993, 1.7880092588117191]}

 Final RMSE for equal weight P is {'Westernwear-Women': 0.8021487772920121, 'I
ndianwear-Women': 0.751406029606777, 'Lingerie&Nightwear-Women': 0.53043786307
35841, 'Footwear-Women': 0.9722708496257133, 'Watches-Women': 0.74578772007143
4, 'Fragrance-Women': 0.7979766706581457, 'Jewellery-Women': 1.021294694615575
4}

 three coun- tries is the distribution ”closest” to equal weight P is Lingerie
&Nightwear-Women,Watches-Women and Indianwear-Women
```

Question#5 discuss your findings?

From my point of veiw,I believe bernford's law are more accurate to present the fact result. On the other hands, I consider that at least in this example,the equal wight distribution is hard to present the fact.