In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
```

In [2]:

```
df = pd.read_csv("diabeties.csv")
```

In [3]:

```
df
```

Out[3]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...	...	...	...	...	...	...	...	...	...
763	10	101	76	48	180	32.9	0.171	63	0
764	2	122	70	27	0	36.8	0.340	27	0
765	5	121	72	23	112	26.2	0.245	30	0
766	1	126	60	0	0	30.1	0.349	47	1
767	1	93	70	31	0	30.4	0.315	23	0

768 rows × 9 columns

In [4]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 9 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   pregnant   768 non-null    int64  
 1   glocose    768 non-null    int64  
 2   bp         768 non-null    int64  
 3   skin       768 non-null    int64  
 4   insulin    768 non-null    int64  
 5   bmi        768 non-null    float64
 6   predigree  768 non-null    float64
 7   age        768 non-null    int64  
 8   target     768 non-null    int64  
dtypes: float64(2), int64(7)
memory usage: 54.1 KB
```

In [5]:

```
df.describe()
```

Out[5]:

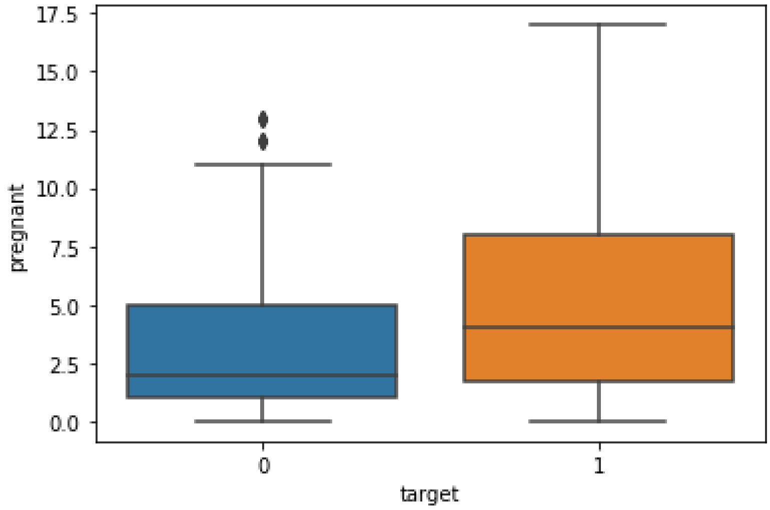
	pregnant	glocose	bp	skin	insulin	bmi	predigree	
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	76
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	0.471876	3
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	0.331329	1
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000	2
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.243750	2
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	0.372500	2
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	0.626250	4
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	8

In [6]:

```
sns.boxplot(data=df,y='pregnant',x='target')
```

Out[6]:

```
<AxesSubplot:xlabel='target', ylabel='pregnant'>
```



In [7]:

```
df[(df.pregnant>11)&(df.target==0)]
```

Out[7]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
28	13	145	82	19	110	22.2	0.245	57	0
86	13	106	72	54	0	36.6	0.178	45	0
274	13	106	70	0	0	34.2	0.251	52	0
333	12	106	80	0	0	23.6	0.137	44	0
358	12	88	74	40	54	35.3	0.378	48	0
436	12	140	85	33	0	37.4	0.244	41	0
518	13	76	60	0	0	32.8	0.180	41	0
582	12	121	78	17	0	26.5	0.259	62	0
744	13	153	88	37	140	40.6	1.174	39	0
745	12	100	84	33	105	30.0	0.488	46	0

In [8]:

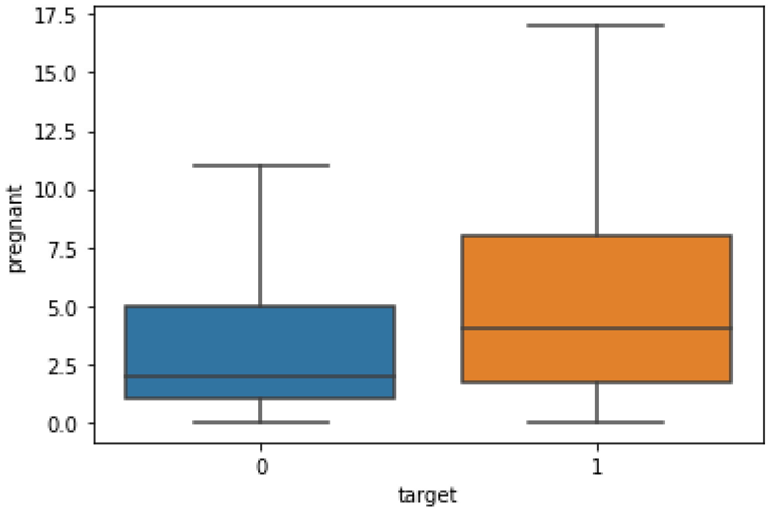
```
df.drop([28,86,274,333,358,436,518,582,744,745],inplace=True)
```

In [9]:

```
sns.boxplot(data=df,y='pregnant',x='target')
```

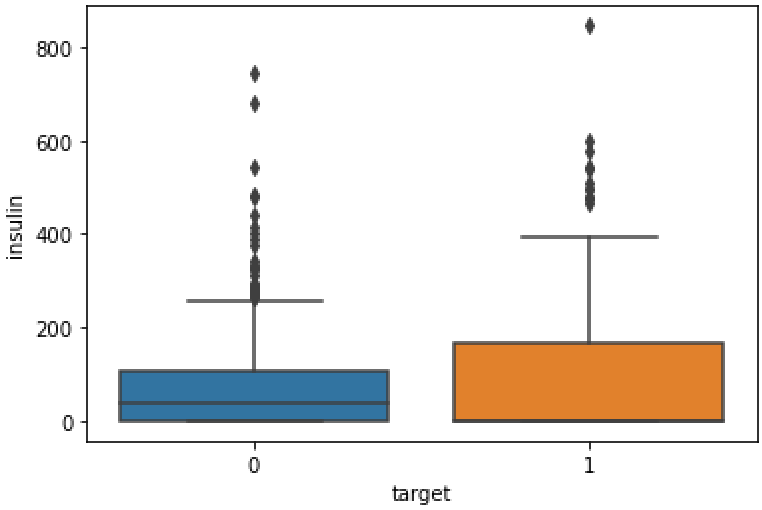
Out[9]:

```
<AxesSubplot:xlabel='target', ylabel='pregnant'>
```



In [10]:

```
sns.boxplot(data=df,x="target",y="insulin")
plt.show()
```



In [11]:

```
df[(df["target"]==0) & (df["insulin"]>240)]
```

Out[11]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
54	7	150	66	42	342	34.7	0.718	42	0
73	4	129	86	20	270	35.1	0.231	23	0
139	5	105	72	29	325	36.9	0.159	28	0
144	4	154	62	31	284	32.8	0.237	23	0
153	1	153	82	42	485	40.6	0.687	23	0
162	0	114	80	34	285	44.2	0.167	27	0
228	4	197	70	39	744	36.7	2.329	31	0
247	0	165	90	33	680	52.3	0.427	23	0
248	9	124	70	33	402	35.4	0.282	34	0
258	1	193	50	16	375	25.9	0.655	24	0
279	2	108	62	10	278	25.3	0.881	22	0
286	5	155	84	44	545	38.7	0.619	34	0
335	0	165	76	43	255	47.9	0.259	26	0
364	4	147	74	25	293	34.9	0.385	30	0
392	1	131	64	14	415	23.7	0.389	21	0
395	2	127	58	24	275	27.7	1.600	25	0
412	1	143	84	23	310	42.4	1.076	22	0
486	1	139	62	41	480	40.7	0.536	21	0
487	0	173	78	32	265	46.5	1.159	58	0
519	6	129	90	7	326	19.6	0.582	60	0
574	1	143	86	30	330	30.1	0.892	23	0
608	0	152	82	39	272	41.5	0.270	27	0
645	2	157	74	35	440	39.4	0.134	30	0
679	2	101	58	17	265	24.2	0.614	23	0
707	2	127	46	21	335	34.4	0.176	22	0
710	3	158	64	13	387	31.2	0.295	24	0
713	0	134	58	20	291	26.4	0.352	21	0

In [12]:

```
df.drop([54,73,139,144,153,162,228,247,248,258,279,286,335,364,392,395,412,486,487,519,574,
```

In [13]:

```
df[(df["target"]==1) & (df["insulin"]>420)]
```

Out[13]:

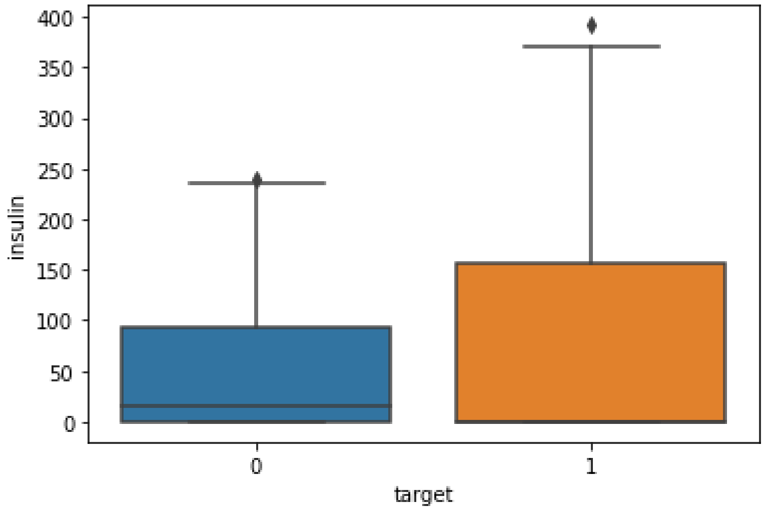
	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
8	2	197	70	45	543	30.5	0.158	53	1
13	1	189	60	23	846	30.1	0.398	59	1
111	8	155	62	26	495	34.0	0.543	46	1
186	8	181	68	36	495	30.1	0.615	60	1
220	0	177	60	29	478	34.6	1.072	21	1
370	3	173	82	48	465	38.4	2.137	25	1
409	1	172	68	49	579	42.4	0.702	28	1
415	3	173	84	33	474	35.7	0.258	22	1
584	8	124	76	24	600	28.7	0.687	52	1
655	2	155	52	27	540	38.7	0.240	25	1
695	7	142	90	24	480	30.4	0.128	43	1
753	0	181	88	44	510	43.3	0.222	26	1

In [14]:

```
df.drop([8,13,111,186,220,370,409,415,584,655,695,753],inplace=True)
```

In [15]:

```
sns.boxplot(data=df,x="target",y="insulin")
plt.show()
```



In [16]:

```
df[(df["target"]==1) & (df["insulin"]>370)]
```

Out[16]:

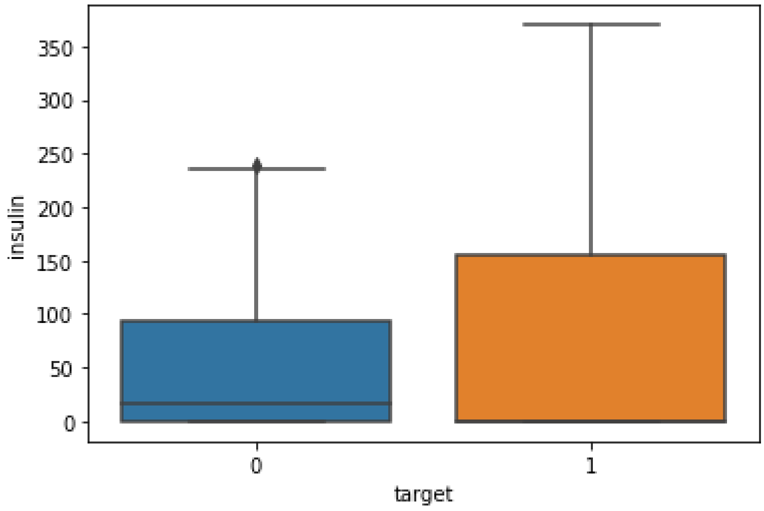
	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
715	7	187	50	33	392	33.9	0.826	34	1

In [17]:

```
df.drop([715],inplace=True)
```

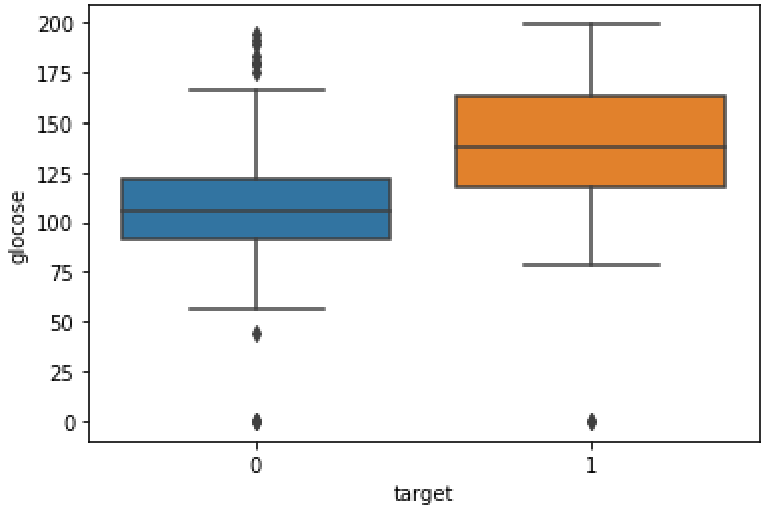
In [18]:

```
sns.boxplot(data=df,x="target",y="insulin")
plt.show()
```



In [19]:

```
sns.boxplot(data=df,x="target",y="glocose")
plt.show()
```



In [20]:

```
df[(df["target"]==0) & (df["glocose"]>174)]
```

Out[20]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
40	3	180	64	25	70	34.0	0.271	26	0
212	7	179	95	31	0	34.2	0.164	60	0
260	3	191	68	15	130	30.9	0.299	34	0
327	10	179	70	0	0	35.1	0.200	37	0
489	8	194	80	0	0	26.1	0.551	67	0
549	4	189	110	31	0	28.5	0.680	37	0
622	6	183	94	0	0	40.8	1.461	45	0
728	2	175	88	0	0	22.9	0.326	22	0

In [21]:

```
df.drop([40,212,260,327,489,549,622,728],inplace=True)
```

In [22]:

```
df[(df["target"]==0) & (df["glocose"]<50)]
```

Out[22]:

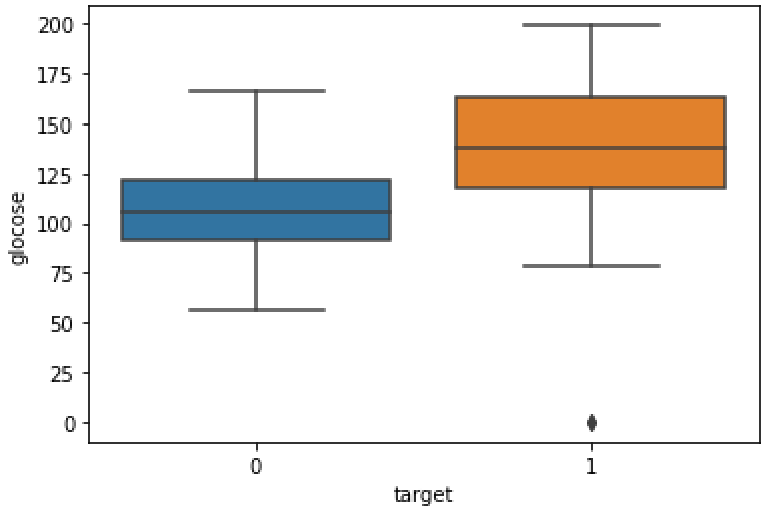
	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
62	5	44	62	0	0	25.0	0.587	36	0
75	1	0	48	20	0	24.7	0.140	22	0
182	1	0	74	20	23	27.7	0.299	21	0
342	1	0	68	35	0	32.0	0.389	22	0

In [23]:

```
df.drop([62,75,182,342],inplace=True)
```

In [24]:

```
sns.boxplot(data=df,x="target",y="glocose")
plt.show()
```



In [25]:

```
df[(df["target"]==1) & (df["glocose"]<50)]
```

Out[25]:

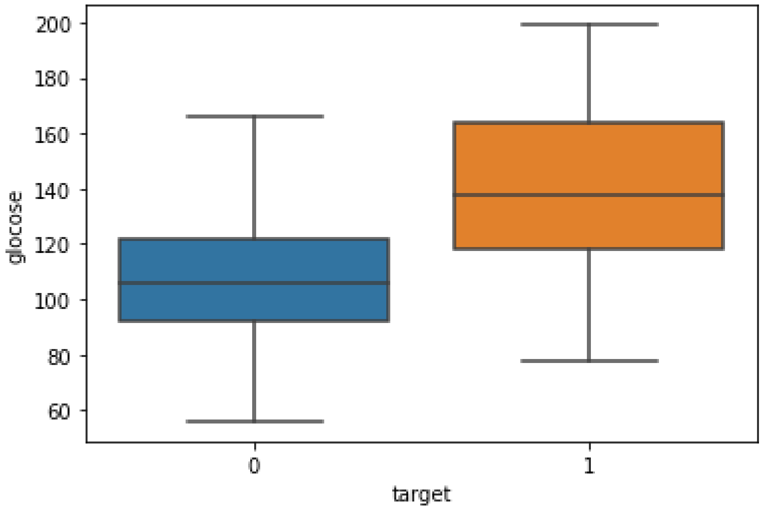
	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
349	5	0	80	32	0	41.0	0.346	37	1
502	6	0	68	41	0	39.0	0.727	41	1

In [26]:

```
df.drop([349,502],inplace=True)
```

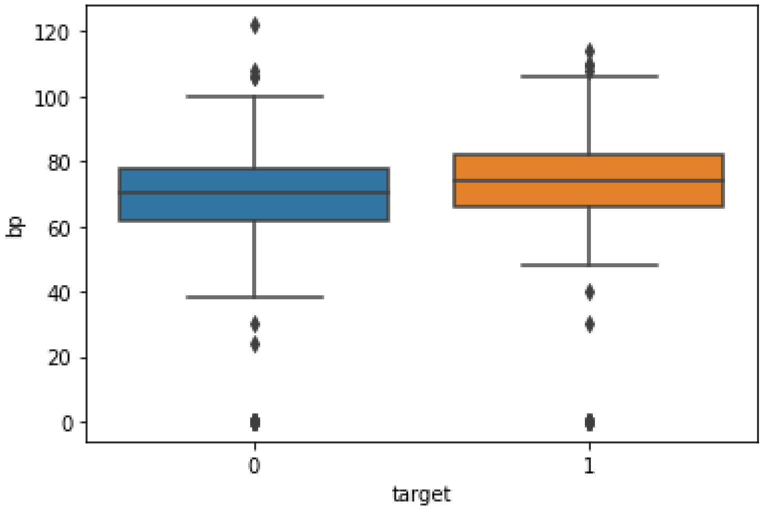
In [27]:

```
sns.boxplot(data=df,x="target",y="glocose")
plt.show()
```



In [28]:

```
sns.boxplot(data=df,x="target",y="bp")
plt.show()
```



In [29]:

```
df[(df["target"]==0) & (df["bp"]>100)]
```

Out[29]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
106	1	96	122	0	0	22.4	0.207	27	0
362	5	103	108	37	0	39.2	0.305	65	0
658	11	127	106	0	0	39.0	0.190	51	0
672	10	68	106	23	49	35.5	0.285	47	0

In [30]:

```
df.drop([106,362,658,672],inplace=True)
```

In [31]:

```
df[(df["target"]==0) & (df["bp"]<38)]
```

Out[31]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
7	10	115	0	0	0	35.3	0.134	29	0
18	1	103	30	38	83	43.3	0.183	33	0
49	7	105	0	0	0	0.0	0.305	24	0
60	2	84	0	0	0	0.0	0.304	21	0
81	2	74	0	0	0	0.0	0.102	22	0
172	2	87	0	23	0	28.9	0.773	25	0
222	7	119	0	0	0	25.2	0.209	37	0
336	0	117	0	0	0	33.8	0.932	44	0
347	3	116	0	0	0	23.5	0.187	23	0
426	0	94	0	0	0	0.0	0.256	25	0
430	2	99	0	0	0	22.2	0.108	23	0
453	2	119	0	0	0	19.6	0.832	72	0
494	3	80	0	0	0	0.0	0.174	22	0
522	6	114	0	0	0	0.0	0.189	26	0
533	6	91	0	0	0	29.8	0.501	31	0
589	0	73	0	0	0	21.1	0.342	25	0
597	1	89	24	19	25	27.8	0.559	21	0
601	6	96	0	0	0	23.7	0.190	28	0
643	4	90	0	0	0	28.0	0.610	31	0
697	0	99	0	0	0	25.0	0.253	22	0
703	2	129	0	0	0	38.5	0.304	41	0

In [32]:

```
df.drop([7,18,49,60,81,172,222,336,347,426,430,453,494,522,533,589,597,601,643,697,703],inp
```

In [33]:

```
df[(df["target"]==1) & (df["bp"]<45)]
```

Out[33]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
4	0	137	40	35	168	43.1	2.288	33	1
15	7	100	0	0	0	30.0	0.484	32	1
78	0	131	0	0	0	43.2	0.270	26	1
125	1	88	30	42	99	55.0	0.496	26	1
193	11	135	0	0	0	52.3	0.578	40	1
261	3	141	0	0	0	30.0	0.761	27	1
266	0	138	0	0	0	36.3	0.933	25	1
269	2	146	0	0	0	27.5	0.240	28	1
300	0	167	0	0	0	32.3	0.839	30	1
332	1	180	0	0	0	43.3	0.282	41	1
357	13	129	0	30	0	39.9	0.569	44	1
435	0	141	0	0	0	42.4	0.205	29	1
468	8	120	0	0	0	30.0	0.183	38	1
484	0	145	0	0	0	44.2	0.630	31	1
535	4	132	0	0	0	32.9	0.302	23	1
604	4	183	0	0	0	28.4	0.212	36	1
619	0	119	0	0	0	32.4	0.141	24	1
706	10	115	0	0	0	0.0	0.261	30	1

In [34]:

```
df.drop([4,15,78,125,193,261,266,269,300,332,357,435,468,484,535,604,619,706],inplace=True)
```

In [35]:

```
df[(df["target"]==1) & (df["bp"]>105)]
```

Out[35]:

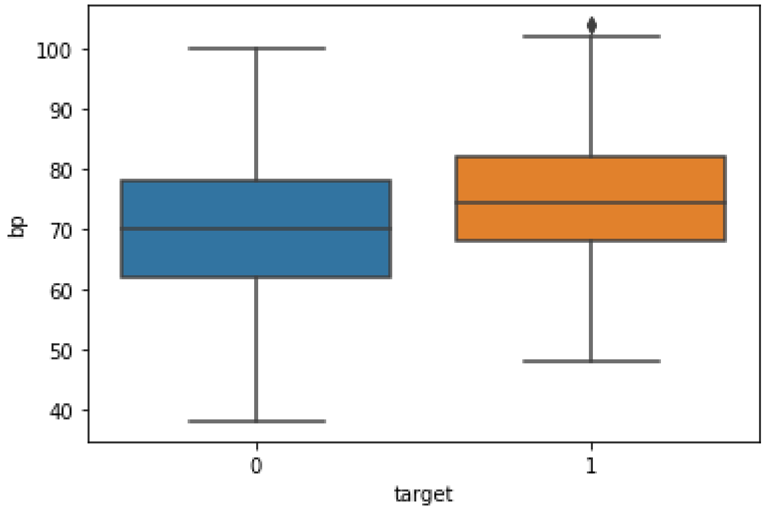
	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
43	9	171	110	24	240	45.4	0.721	54	1
84	5	137	108	0	0	48.8	0.227	37	1
177	0	129	110	46	130	67.1	0.319	26	1
662	8	167	106	46	231	37.6	0.165	43	1
691	13	158	114	0	0	42.3	0.257	44	1

In [36]:

```
df.drop([43,84,177,662,691],inplace=True)
```

In [37]:

```
sns.boxplot(data=df,x="target",y="bp")
plt.show()
```

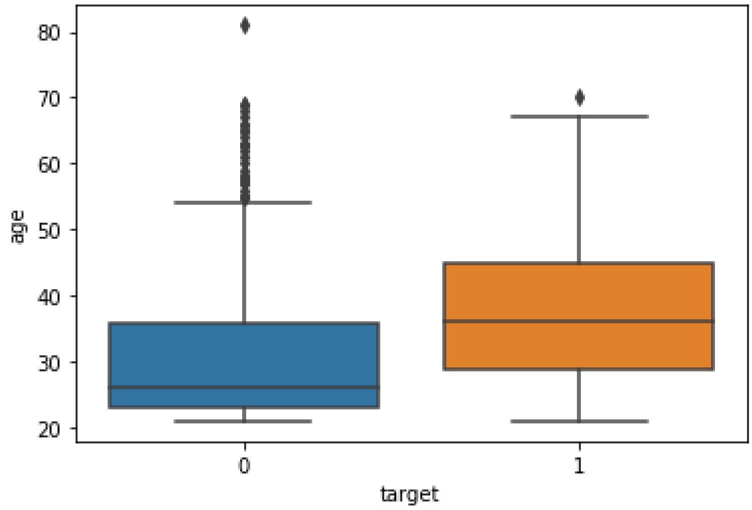


In [38]:

```
sns.boxplot(data=df,x='target',y='age')
```

Out[38]:

```
<AxesSubplot:xlabel='target', ylabel='age'>
```



In [39]:

```
df[(df.age>55)&(df.target==0)]
```

Out[39]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
12	10	139	80	0	0	27.1	1.441	57	0
30	5	109	75	26	0	36.0	0.546	60	0
123	5	132	80	0	0	26.8	0.186	69	0
148	5	147	78	0	0	33.7	0.218	65	0
223	7	142	60	33	190	28.8	0.687	61	0
263	3	142	80	15	0	32.4	0.200	63	0
278	5	114	74	0	0	24.9	0.744	57	0
294	0	161	50	0	0	21.9	0.254	65	0
299	8	112	72	0	0	23.6	0.840	58	0
344	8	95	72	0	0	36.8	0.485	57	0
361	5	158	70	0	0	29.8	0.207	63	0
456	1	135	54	0	0	26.7	0.687	62	0
459	9	134	74	33	60	25.9	0.460	81	0
475	0	137	84	27	0	27.3	0.231	59	0
479	4	132	86	31	0	28.0	0.419	63	0
495	6	166	74	0	0	26.6	0.304	66	0
509	8	120	78	0	0	25.0	0.409	64	0
512	9	91	68	0	0	24.2	0.200	58	0
537	0	57	60	0	0	21.7	0.735	67	0
552	6	114	88	0	0	27.8	0.247	66	0
557	8	110	76	0	0	27.8	0.237	58	0
674	8	91	82	0	0	35.6	0.587	68	0
684	5	136	82	0	0	0.0	0.640	69	0
717	10	94	72	18	0	23.1	0.595	56	0
763	10	101	76	48	180	32.9	0.171	63	0

In [40]:

```
df.drop([12,30,123,148,223,263,278,294,299,344,361,456,459,475,479,495,509,512,537,552,557,
```

In [41]:

```
df[(df.age>65)&(df.target==1)]
```

Out[41]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
221	2	158	90	0	0	31.6	0.805	66	1
363	4	146	78	0	0	38.5	0.520	67	1
666	4	145	82	18	0	32.5	0.235	70	1
759	6	190	92	0	0	35.5	0.278	66	1

In [42]:

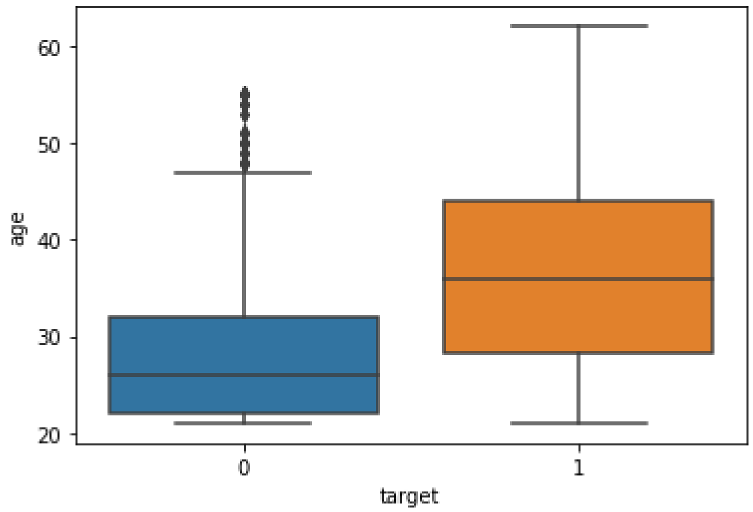
```
df.drop([221,363,666,759],inplace=True)
```

In [43]:

```
sns.boxplot(data=df,x='target',y='age')
```

Out[43]:

```
<AxesSubplot:xlabel='target', ylabel='age'>
```



In [44]:

```
df.describe()
```

Out[44]:

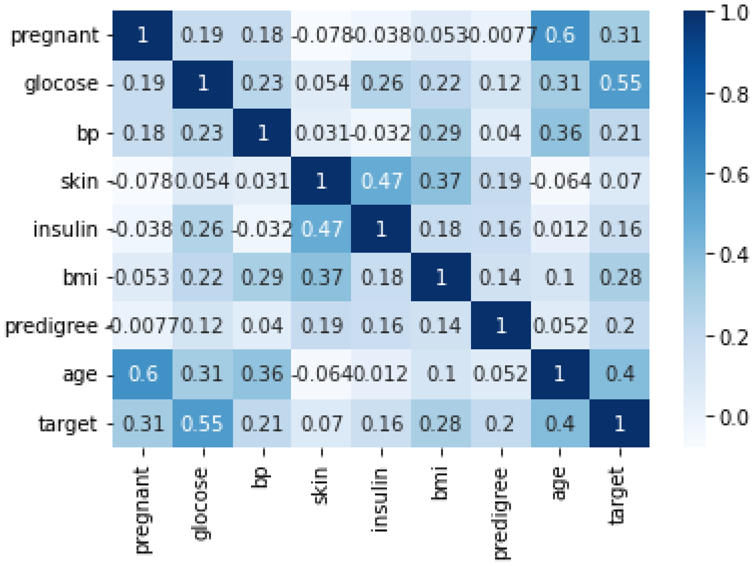
	pregnant	glocose	bp	skin	insulin	bmi	predigree	
count	627.000000	627.000000	627.000000	627.000000	627.000000	627.000000	627.000000	62
mean	3.661882	119.153110	71.917065	21.497608	67.540670	32.137959	0.470439	3
std	3.204509	29.317293	11.206006	15.584088	80.135565	7.021421	0.316019	
min	0.000000	56.000000	38.000000	0.000000	0.000000	0.000000	0.078000	2
25%	1.000000	99.000000	64.000000	0.000000	0.000000	27.450000	0.248500	2
50%	3.000000	114.000000	72.000000	24.000000	44.000000	32.000000	0.380000	2
75%	6.000000	136.000000	80.000000	32.000000	120.000000	36.400000	0.626500	3
max	17.000000	199.000000	104.000000	99.000000	370.000000	59.400000	2.420000	6

In [45]:

```
sns.heatmap(df.corr(),annot=True,cmap="Blues")
```

Out[45]:

```
<AxesSubplot:>
```



	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
pregnant	1	0.19	0.18	-0.078	-0.038	-0.053	-0.0077	0.6	0.31
glocose	0.19	1	0.23	0.054	0.26	0.22	0.12	0.31	0.55
bp	0.18	0.23	1	0.031	-0.032	0.29	0.04	0.36	0.21
skin	-0.078	0.054	0.031	1	0.47	0.37	0.19	-0.064	0.07
insulin	-0.038	0.26	-0.032	0.47	1	0.18	0.16	0.012	0.16
bmi	-0.053	0.22	0.29	0.37	0.18	1	0.14	0.1	0.28
predigree	-0.0077	0.12	0.04	0.19	0.16	0.14	1	0.052	0.2
age	0.6	0.31	0.36	-0.064	0.012	0.1	0.052	1	0.4
target	0.31	0.55	0.21	0.07	0.16	0.28	0.2	0.4	1

In [46]:

```
features = df.iloc[:,0:-1]
target = df.iloc[:,-1]
```

In [47]:

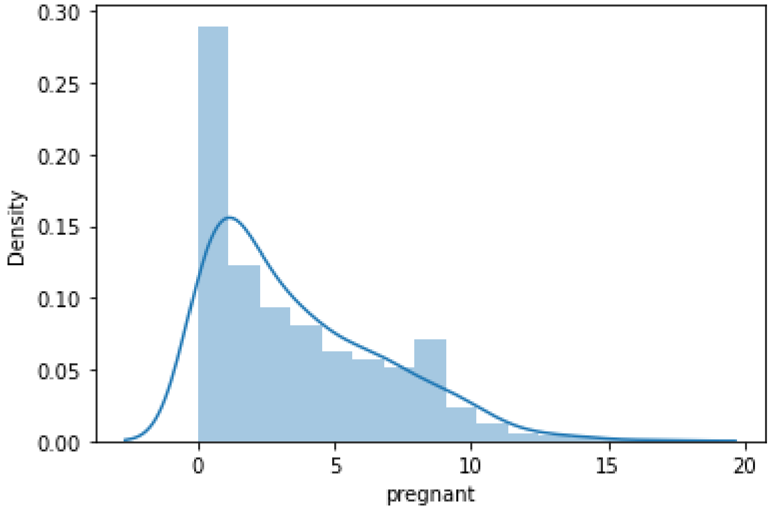
```
from scipy.stats import skew
```

In [48]:

```
for col in features:
    print(col)
    print(skew(features[col]))

    plt.figure()
    sns.distplot(features[col])
    plt.show()
```

```
pregnant
0.9515941317169568
```



```
glocose
```

In [49]:

```
pd.concat([features,target],axis=1).corr()["target"].sort_values()
```

Out[49]:

```
skin         0.069659
insulin      0.160167
predigree    0.199401
bp           0.210557
bmi          0.284899
pregnant     0.311659
age          0.397477
glocose      0.552735
target       1.000000
Name: target, dtype: float64
```

In [50]:

```
pd.concat([features,target],axis=1).corr()
```

Out[50]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age
pregnant	1.000000	0.186129	0.180450	-0.078226	-0.038191	0.053187	-0.007658	0.597530
glocose	0.186129	1.000000	0.232082	0.054090	0.255597	0.216248	0.116560	0.308083
bp	0.180450	0.232082	1.000000	0.030935	-0.032258	0.288415	0.039711	0.363539
skin	-0.078226	0.054090	0.030935	1.000000	0.473586	0.373287	0.189137	-0.064468
insulin	-0.038191	0.255597	-0.032258	0.473586	1.000000	0.176892	0.163731	0.012422
bmi	0.053187	0.216248	0.288415	0.373287	0.176892	1.000000	0.139284	0.104701
predigree	-0.007658	0.116560	0.039711	0.189137	0.163731	0.139284	1.000000	0.052248
age	0.597530	0.308083	0.363539	-0.064468	0.012422	0.104701	0.052248	1.000000
target	0.311659	0.552735	0.210557	0.069659	0.160167	0.284899	0.199401	0.397477

In [51]:

```
pd.concat([features,target],axis=1).corr().style.background_gradient()
```

Out[51]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age
pregnant	1.000000	0.186129	0.180450	-0.078226	-0.038191	0.053187	-0.007658	0.597530
glocose	0.186129	1.000000	0.232082	0.054090	0.255597	0.216248	0.116560	0.308083
bp	0.180450	0.232082	1.000000	0.030935	-0.032258	0.288415	0.039711	0.363539
skin	-0.078226	0.054090	0.030935	1.000000	0.473586	0.373287	0.189137	-0.064468
insulin	-0.038191	0.255597	-0.032258	0.473586	1.000000	0.176892	0.163731	0.012422
bmi	0.053187	0.216248	0.288415	0.373287	0.176892	1.000000	0.139284	0.104701
predigree	-0.007658	0.116560	0.039711	0.189137	0.163731	0.139284	1.000000	0.052248
age	0.597530	0.308083	0.363539	-0.064468	0.012422	0.104701	0.052248	1.000000
target	0.311659	0.552735	0.210557	0.069659	0.160167	0.284899	0.199401	0.397477

In [52]:

```
features["insulin"]=np.sqrt(features["insulin"])
```

In [53]:

```
skew(features["insulin"])
```

Out[53]:

```
0.2679666871964709
```

In [54]:

```
pd.concat([features,target],axis=1).corr().style.background_gradient()
```

Out[54]:

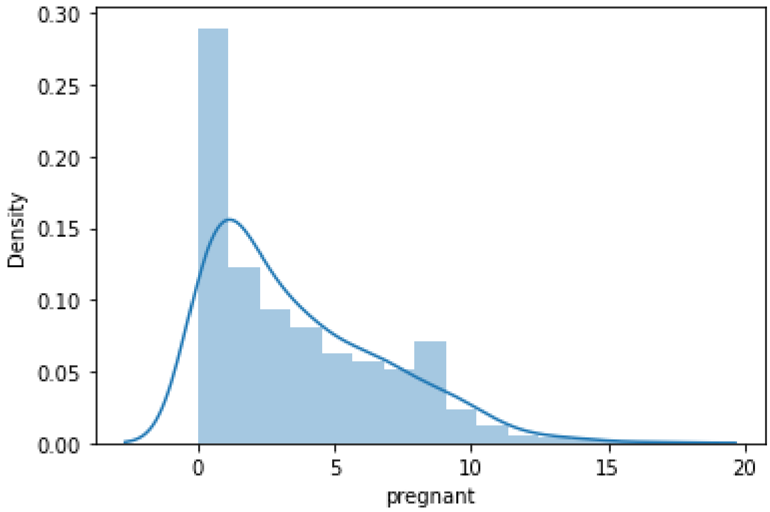
	pregnant	glocose	bp	skin	insulin	bmi	predigree	age
pregnant	1.000000	0.186129	0.180450	-0.078226	-0.098316	0.053187	-0.007658	0.597530
glocose	0.186129	1.000000	0.232082	0.054090	0.150652	0.216248	0.116560	0.308083
bp	0.180450	0.232082	1.000000	0.030935	-0.081498	0.288415	0.039711	0.363539
skin	-0.078226	0.054090	0.030935	1.000000	0.514264	0.373287	0.189137	-0.064468
insulin	-0.098316	0.150652	-0.081498	0.514264	1.000000	0.133850	0.168490	-0.069922
bmi	0.053187	0.216248	0.288415	0.373287	0.133850	1.000000	0.139284	0.104701
predigree	-0.007658	0.116560	0.039711	0.189137	0.168490	0.139284	1.000000	0.052248
age	0.597530	0.308083	0.363539	-0.064468	-0.069922	0.104701	0.052248	1.000000
target	0.311659	0.552735	0.210557	0.069659	0.066823	0.284899	0.199401	0.397477

In [55]:

```
for col in features:
    print(col)
    print(skew(features[col]))

    plt.figure()
    sns.distplot(features[col])
    plt.show()
```

```
pregnant
0.9515941317169568
```



```
glucose
```

In [56]:

```
features.describe()
```

Out[56]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	
count	627.000000	627.000000	627.000000	627.000000	627.000000	627.000000	627.000000	62
mean	3.661882	119.153110	71.917065	21.497608	5.797923	32.137959	0.470439	3
std	3.204509	29.317293	11.206006	15.584088	5.829147	7.021421	0.316019	
min	0.000000	56.000000	38.000000	0.000000	0.000000	0.000000	0.078000	2
25%	1.000000	99.000000	64.000000	0.000000	0.000000	27.450000	0.248500	2
50%	3.000000	114.000000	72.000000	24.000000	6.633250	32.000000	0.380000	2
75%	6.000000	136.000000	80.000000	32.000000	10.954451	36.400000	0.626500	3
max	17.000000	199.000000	104.000000	99.000000	19.235384	59.400000	2.420000	6

In [57]:

```
from sklearn.model_selection import train_test_split

xtrain,xtest,ytrain,ytest = train_test_split(features,target,test_size=0.4,random_state=1)
```

In [58]:

```
from sklearn.linear_model import LogisticRegression

logreg = LogisticRegression()
logreg.fit(xtrain,ytrain)
ypred = logreg.predict(xtest)
```

In [59]:

```
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report

ac = accuracy_score(ytest,ypred)
cm = confusion_matrix(ytest,ypred)
cr = classification_report(ytest,ypred)

print(f"Accuracy Score:- {ac}\nConfusion Matrix:-\n{cm}\nClassificationReport:-\n{cr}")
```

```
Accuracy Score:- 0.8047808764940239
Confusion Matrix:-
[[139  22]
 [ 27  63]]
ClassificationReport:-
              precision    recall  f1-score   support

           0       0.84      0.86      0.85       161
           1       0.74      0.70      0.72        90

    accuracy                           0.80       251
   macro avg       0.79      0.78      0.79       251
weighted avg       0.80      0.80      0.80       251
```

In [60]:

```
train = logreg.score(xtrain,ytrain)
```

In [61]:

```
test = logreg.score(xtest,ytest)
```

In [62]:

```
print(f"Training Score:-{train}\nTesting Score:-{test}")
```

```
Training Score:-0.8085106382978723
Testing Score:-0.8047808764940239
```

In [63]:

```
df.columns
```

Out[63]:

```
Index(['pregnant', 'glocose', 'bp', 'skin', 'insulin', 'bmi', 'predigree',
       'age', 'target'],
      dtype='object')
```

In [64]:

```
def predictdiabetes(preg,glc,bp,skin,insulin,bmi,predi,age):
    newobs = [[preg,glc,bp,skin,insulin,bmi,predi,age]]
    ypred = logreg.predict(newobs)[0]

    return ypred
```

In [65]:

```
df.head(6)
```

Out[65]:

	pregnant	glocose	bp	skin	insulin	bmi	predigree	age	target
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
5	5	116	74	0	0	25.6	0.201	30	0
6	3	78	50	32	88	31.0	0.248	26	1

In [66]:

```
predictdiabetes(6,148,72,35,0,33.6,0.627,50)
```

Out[66]:

```
1
```

In [67]:

```
predictdiabetes(1,85,66,29,0,26,0.3,31)
```

Out[67]:

```
0
```

In [68]:

```
predictdiabetes(8,183,64,0,0,23.3,0.672,32)
```

Out[68]:

```
1
```

In [69]:

```
predictdiabetes(1,89,66,23,94,28.1,0.167,21)
```

Out[69]:

```
1
```

In [70]:

```
predictdiabetes(5,116,74,0,0,25.6,0.201,30)
```

Out[70]:

```
0
```

In [71]:

```
predictdiabetes(3,78,50,32,88,31.0,0.248,26)
```

Out[71]:

```
1
```