BIG DATA TA2 PROJECT

NAME : ADITI KULKARNI

ENROLLMENT NUMBER : 101CTBMCS2122018

PROJECT DEFINITION

- The objective of this project is to analyze the dataset from National Institute of Diabetes and Digestive and Kidney Diseases.
- Analysis is done to understand whether a patient has diabetes or not based on certain diagnostic measurements included in the dataset
- All patients in the dataset are female with age 21 years or more.

DATASET FEATURES

1. Pregnancies: Number of times pregnant
2. Glucose: Plasma glucose concentration 2 hours after an oral glucose tolerance test
3. BloodPressure: Diastolic blood pressure (mm Hg)
4. SkinThickness: Triceps skin fold thickness (mm)
5. Insulin: 2-Hour serum insulin (mu U/ml)
6. BMI: Body mass index (weight in kg/(height in m^2))
7. DiabetesPedigreeFunction: Diabetes pedigree function
8. Age: Age (years)
9. Outcome: Diabetic or not (0 for no, 1 for yes)

TOOL USED : PANDAS

- To analyze the given dataset and find out results for the questions, pandas library of python is used.
- This library is rich of functions to deal with large datasets and can work with data in many formats like csv, excel, tsv, txt, etc.
- Pandas makes it easy to query for data with certain conditions and return output in required formats including options for graphs, charts, etc.
- Pandas is an open-source data analysis and manipulation tool built on top of the Python programming language. It is essential for data cleaning, transformation, and analysis.

OBJECTIVE

This project aims to derive meaningful insights and answers to specific questions from the dataset:

1. Identify the number of people below the age of 30 who are diabetic.
2. Determine how many women chose not to be pregnant while being diabetic.
3. Find the greatest number of pregnancies a woman had while being diabetic.
4. Explore the number of people with a Diabetes pedigree function below 0.250 who are diabetic.
5. Investigate how many people with insulin levels below 200 are diabetic.
6. Understand the overall distribution of diabetic and non-diabetic individuals in the dataset.
7. Determine the percentage of women under 25 years old who have had at least one pregnancy.
8. Identify the number of women with a skin thickness above 29 who are diabetic.
9. Explore the number of people with a glucose level above 190.
10. Investigate the number of diabetics with a BMI over 30.
11. Determine the number of diabetics with a blood pressure of less than 80.

EXECUTION

Loading Dataset in Pandas

Here the pandas and matplotlib library is being imported for data analysis and plotting.

Using the pd.read_csv() function the dataset is being loaded and stored as a dataframe type.

The head function is displaying the top 5 records from the dataset

The len(df) indicates that there are in total 768 rows in the dataset.

```
[5]  import pandas as pd
     import matplotlib.pyplot as plt

     df = pd.read_csv('diabetes.csv')
     df.head()
```

	Pregnancies	Glucose	BloodPressure	Triceps	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
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

```
len(df)
```

```
768
```

Data Cleaning

The missing values are being detected by isna() function which outputs the number of null values in all the columns.

If there are null values then they are dropped by dropna() function.

The inplace=True ensures that after the function the old dataframe is updated with result.

Data Cleaning

```
# Check for missing values
missing_values = df.isna().sum()  # to count missing values in each column
print(missing_values)

# Remove missing values
df.dropna(inplace=True)

# Reset index
df.reset_index(drop=True, inplace=True)  # this is done to make the numbering proper after removing the middle null values
```

```
Pregnancies                 0
Glucose                     0
BloodPressure               0
Triceps                     0
Insulin                     0
BMI                         0
DiabetesPedigreeFunction    0
Age                         0
Outcome                     0
dtype: int64
```

Q1. How many people below the age of 30 are diabetic in the dataset?

The df.query() function is used to return rows with value of column age less than 30 and Outcome = 1 meaning that they are diabetic.

How many people below the age of 30 are diabetic in the dataset?

```
[28] result1 = df.query('Age < 30 & Outcome == 1')  # returns the rows having age less than 30
     result1.head()
     print("Number of people below the age of 30 diabetic in the dataset are = ",len(result1))
```

```
Number of people below the age of 30 diabetic in the dataset are =  84
```

Inference = There are very few people diabetic people with age < 30 since total diabetic are 268. Thus, young people are at low risk of diabetes.

Q2. How many women chose NOT to be pregnant while being a diabetic?

The df.query() function is used to return rows with 0 Pregnancies but still Outcome = 1

How many women chose NOT to be pregnant while being a diabetic?

```
[8]  result2 = df.query('Pregnancies == 0 & Outcome == 1')  # returns the rows with 0 pregnancies but are diabetic
     result2.head()
     print("Number of women who chose NOT to be pregnant while being diabetic = ", len(result2))
```

```
Number of women who chose NOT to be pregnant while being diabetic =  38
```

Inference = There are very few people diabetic people with 0 pregnancies. Thus, very few people took a calculated and thoughtful decision since very few are well educated about this disease. Diabetes mostly spreads genetically which could create risk of diabetes for the child as well. So more people should be aware about this.

Q3. What is the greatest number of pregnancies a woman had while being a diabetic?

Here first the dataframe is sorted with number of pregnancies in descending order. Then df.query() function is used to return rows with Outcome 1. So the final dataframe is sorted with all diabetic patients and result is max of this dataframe.

What is the greatest number of pregnancies a woman had while being a diabetic?

```
[9]  result3 = df.sort_values(by='Pregnancies', ascending=False).query("Outcome == 1")  # returns dataframe with diabetic women sorted in
                                                                                        # descending orders by their pregnancy number
     result3.head()

     max_pregnancies = max(result3.Pregnancies)  # returns greatest number of pregnancies
     print("Greatest number of pregnancies a women had while being a diabetic = ", max_pregnancies)
```

```
Greatest number of pregnancies a women had while being a diabetic =  17
```

Inference = The maximum number of pregnancies are very high. This might indicate that becoming more times pregnant can make the mother weak and result in some disease like diabetes.

Q4. How many people with a Diabetes pedigree function below 0.250 are diabetic?

Then df.query() function is used to return rows with Diabetes Pedigree Function less than 0.25 and Outcome = 1 i.e., diabetic.

How many people with a Diabetes pedigree function below 0.250 are diabetic?

```
[10] result4 = df.query('DiabetesPedigreeFunction < 0.250 & Outcome == 1')
     result4.head()
     print("Number of people with Diabetes pedigree function below 0.250 who are diabetic = ", len(result4))
```

```
Number of people with Diabetes pedigree function below 0.250 who are diabetic =  52
```

Inference = The diabetic pedigree function scores the probability of diabetes based on family history, with a realistic range of 0.08 to 2.42. The lesser the value the lesser are the number of diabetic family members. So, below 0.25 there are only few diabetic females since for 0.25 below the family history for diabetes is also very less, validating the diabetes pedigree function.

Q5. How many people with Insulin levels lesser than 200 are diabetic?

Then df.query() function is used to return rows with Insulin Levels less than 200 and Outcome = 1 i.e., diabetic.

How many people with Insulin levels lesser than 200 are diabetic?

```
[11] result5 = df.query('Insulin < 200 & Outcome == 1')
     result5.head()
     print("Number of people with insulin levels lesser than 200 are diabetic = ", len(result5))
```

```
Number of people with insulin levels lesser than 200 are diabetic =  221
```

Inference = The insulin levels mentioned here are fasting timed which means level after 2hr of fasting. Normally it should be < 25 mIU/L. Here less than 200 mIU/L are higher out of 268 diabetic people which indicate that insulin levels from 25 to 200 means the person is diabetic.

Q6. Overall, how many people in the dataset are diabetic and non-diabetic?

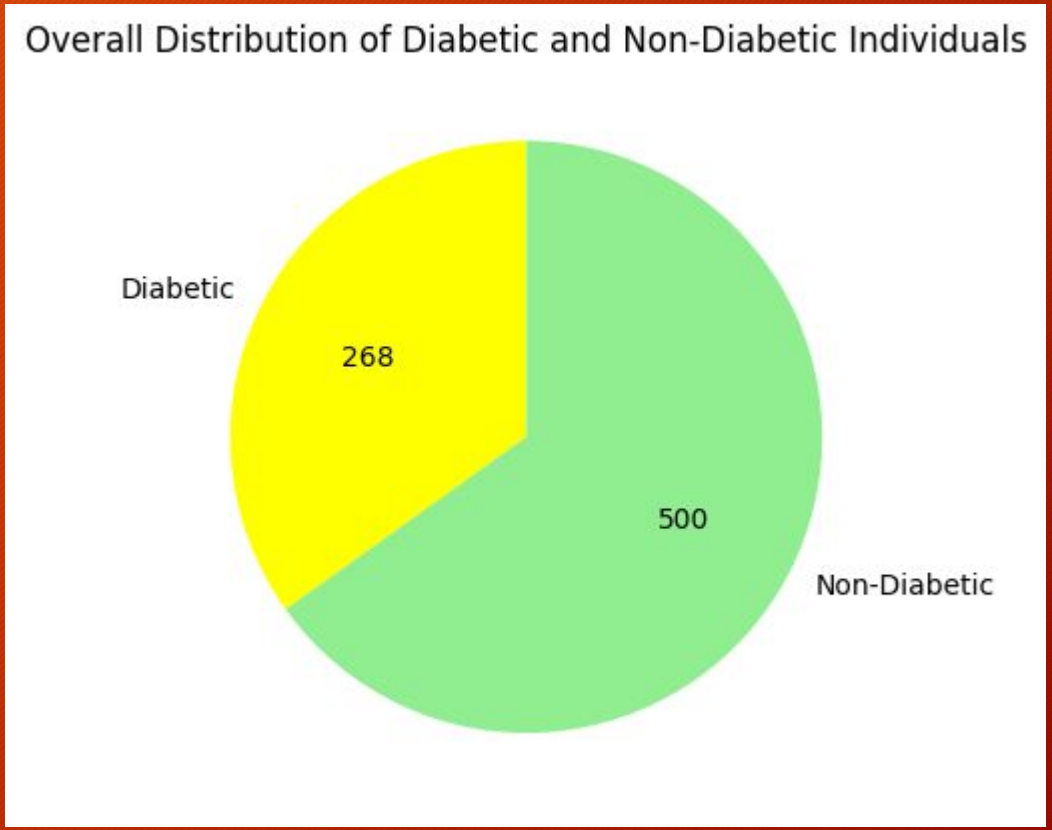
Overall, how many people in the dataset are diabetic and non-diabetic?

```
[12] result6 = df.query('Outcome == 1')
     diabetic_count = len(result6)
     result6 = df.query('Outcome == 0')
     non_diabetic_count = len(result6)
     print("Number of Diabetic People = ", diabetic_count)
     print("Number of Non - Diabetic People = ", non_diabetic_count)
```

```
Number of Diabetic People =  268
Number of Non - Diabetic People =  500
```

```
# Visualization: Pie Chart
labels = ['Diabetic', 'Non-Diabetic']
sizes = [diabetic_count, non_diabetic_count]
colors = ['yellow', 'lightgreen']

# plt.figure(figsize=(8, 8))
plt.pie(sizes, labels=labels, autopct=lambda p: f'{int(p * sum(sizes) / 100)}', colors=colors, startangle=90)
plt.title('Overall Distribution of Diabetic and Non-Diabetic Individuals')
plt.show()
```



Q7. What percentage of women who are less than 25 years old, have had at least one pregnancy?

Then df.query() function is used to return rows with Age less than 25 years and 1 or more pregnancies. Then the percentage is calculated.

What percentage of women who are less than 25 years old, have had at least one pregnancy?

```
[14] result7 = df.query('Age < 25 & Pregnancies >= 1')
     percentage = (len(result7)/len(df))*100
     print("Percentage of women less than 25 years old with 1 or more pregnancies = ", round(percentage, 2), "%")
```

```
Percentage of women less than 25 years old with 1 or more pregnancies =  21.35 %
```

Inference = There are very few females i.e., only 21.35% females who get pregnant at this young age of 21-25 years.

Q8. How many women with a skin thickness of above 29 are diabetic?

Then df.query() function is used to return rows with Skin thickness above 29 and Outcome = 1 i.e., Diabetic.

How many women with a skin thickness of above 29 are diabetic?

```
[15] result8 = df.query('Triceps > 29 & Outcome == 1')
     result8.head()
     print("Number of women with a skin thickness or above 29mm who are diabetic = ", len(result8))
```

```
Number of women with a skin thickness or above 29mm who are diabetic =  117
```

Inference = Out of 268, 117 women are diabetic with tricep skin fold thickness above 29, which is a quite a large number almost 50%. The normal skin thickness must be 23mm for women, so greater than this indicates fat presence. This maybe is affection the sugar levels and insulin generation resulting in diabetes.

Q9. How many people have a glucose level above 190?

Then df.query() function is used to return rows with Glucose level above 190.

How many people have a glucose level above 190?

```
[16] result9 = df.query('Glucose > 190')
     print("Number of people with glucose level above 190 = ", len(result9))
```

```
Number of people with glucose level above 190 =  17
```

Inference = The normal range for glucose level is 70-99. Here it indicates that there are very few people with very high glucose levels.

Q10. How many diabetics have a BMI of over 30?

Then df.query() function is used to return rows with Outcome =1 and BMI greater than 30.

How many diabetics have a BMI of over 30?

```
[17] result10 = df.query('BMI > 30 & Outcome == 1')
     print("Number of diabetics having BMI over 30 = ", len(result10))
```

```
Number of diabetics having BMI over 30 =  215
```

Inference = There are very large number of women who are obese since BMI > 30 and are diabetic. Since out of 268 diabetic people, 215 are obese. So, more fat in body can be a strong reason for diabetes.

Q11. How many diabetics have a blood pressure of less than 80?

Then df.query() function is used to return rows with Outcome =1 and blood pressure less than 80.

How many diabetics have a blood pressure of less than 80?

```
[18] result11 = df.query('BloodPressure < 80 & Outcome == 1')
     print("Number of diabetics with blood pressure < 80 = ", len(result11))
```

```
Number of diabetics with blood pressure < 80 =  178
```

Inference = Blood pressure less than 80 indicates low blood pressure and since almost half of the women are having low blood pressure and are diabetic since total diabetic are 268, this indicates low blood pressure can be a reason contributing to diabetes.

CONCLUSION

From the above diagnostic questions and their results, we can infer that:

- Young people are at lower risk of diabetes
- Family history contributes for a patient to be diabetic
- More number of pregnancies in women may cause diabetes and must be avoided since diabetes can be passed genetically to the child.
- Insulin levels, glucose levels and blood pressure helps to indicate diabetes.
- Obesity increases the risk of being diabetic.

THANK YOU