2. A detailed explanation of the attributes of the data.
• Describe if the attributes are discrete/continuous, Nominal/Ordinal/Interval/Ratio,
• Give an account of whether there are data issues (i.e. missing values or corrupted data) and describe them if so.
• Include basic summary statistics of the attributes.

If your data set contains many similar attributes, you may restrict yourself to describing a few representative features (apply common sense).

Attributes

sbp-systolic blood pressure-discrete-ratio data
tobacco-cumulative tobacco (kg)-continuous-ratio data
ldl-low density lipoprotein cholesterol-continuous- ratio data
adiposity-continuous-ratio data
famhist-family history heart disease-Nominal
typea-type A behaviors- discrete-ratio data
obesity-continuous-ratio data
alcohol-current alcohol consumption-continuous- ratio data
age-discrete- Ratio data
chd-coronary heart disease (Trevor Hastie, u.d.)-Nomial

Basic Summary

Basic Summary

```
In [8]: df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 462 entries, 0 to 461
Data columns (total 10 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   sbp        462 non-null    int64
 1   tobacco    462 non-null    float64
 2   ldl        462 non-null    float64
 3   adiposity  462 non-null    float64
 4   famhist    462 non-null    object
 5   typea      462 non-null    int64
 6   obesity    462 non-null    float64
 7   alcohol    462 non-null    float64
 8   age        462 non-null    int64
 9   chd        462 non-null    int64
dtypes: float64(5), int64(4), object(1)
memory usage: 36.2+ KB
```

Basic Statistics

Attribute Statistics

```
In [5]: df1.describe(include='all')
```

Out[5]:

	sbp	tobacco	ldl	adiposity	famhist	typea	obesity	alcohol	age	chd
count	462.000000	462.000000	462.000000	462.000000	462	462.000000	462.000000	462.000000	462.000000	462.000000
unique	NaN	NaN	NaN	NaN	2	NaN	NaN	NaN	NaN	NaN
top	NaN	NaN	NaN	NaN	Absent	NaN	NaN	NaN	NaN	NaN
freq	NaN	NaN	NaN	NaN	270	NaN	NaN	NaN	NaN	NaN
mean	138.326840	3.635649	4.740325	25.406732	NaN	53.103896	26.044113	17.044394	42.816017	0.346320
std	20.496317	4.593024	2.070909	7.780699	NaN	9.817534	4.213680	24.481059	14.608956	0.476313
min	101.000000	0.000000	0.980000	6.740000	NaN	13.000000	14.700000	0.000000	15.000000	0.000000
25%	124.000000	0.052500	3.282500	19.775000	NaN	47.000000	22.985000	0.510000	31.000000	0.000000
50%	134.000000	2.000000	4.340000	26.115000	NaN	53.000000	25.805000	7.510000	45.000000	0.000000
75%	148.000000	5.500000	5.790000	31.227500	NaN	60.000000	28.497500	23.892500	55.000000	1.000000
max	218.000000	31.200000	15.330000	42.490000	NaN	78.000000	46.580000	147.190000	64.000000	1.000000

Missing Values

Checking for Missing Values

```
In [6]: for i in range(10):
            print(df1.iloc[:,[i]].isnull().values.any())
        #No Missing Values
```

```
False
False
False
False
False
False
False
False
False
False
```

No missing values