

```

...      id gender age hypertension heart_disease ever_married \
0 9046   Male 67.0          0           1        Yes
1 51676 Female 61.0          0           0        Yes
2 31112   Male 80.0          0           1        Yes
3 60182 Female 49.0          0           0        Yes
4 1665 Female 79.0          1           0        Yes

      work_type Residence_type avg_glucose_level   bmi smoking_status \
0      Private          Urban            228.69 36.6 formerly smoked
1  Self-employed        Rural            202.21   NaN never smoked
2      Private          Rural            105.92 32.5 never smoked
3      Private          Urban            171.23 34.4      smokes
4  Self-employed        Rural            174.12 24.0 never smoked

```

```

stroke
0      1
1      1
2      1
3      1
4      1

```

After Encoding:

	gender	age	hypertension	heart_disease	ever_married	work_type
0	1	67.0	0	1	1	2
1	0	61.0	0	0	1	3
2	1	80.0	0	1	1	2
3	0	49.0	0	0	1	2
4	0	79.0	1	0	1	3

	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	1	228.69	36.600000	1	1
1	0	202.21	28.893237	2	1
2	0	105.92	32.500000	2	1
3	1	171.23	34.400000	3	1
4	0	174.12	24.000000	2	1

```

===== DECISION TREE RESULTS =====
Accuracy: 0.9383561643835616

```

Confusion Matrix:

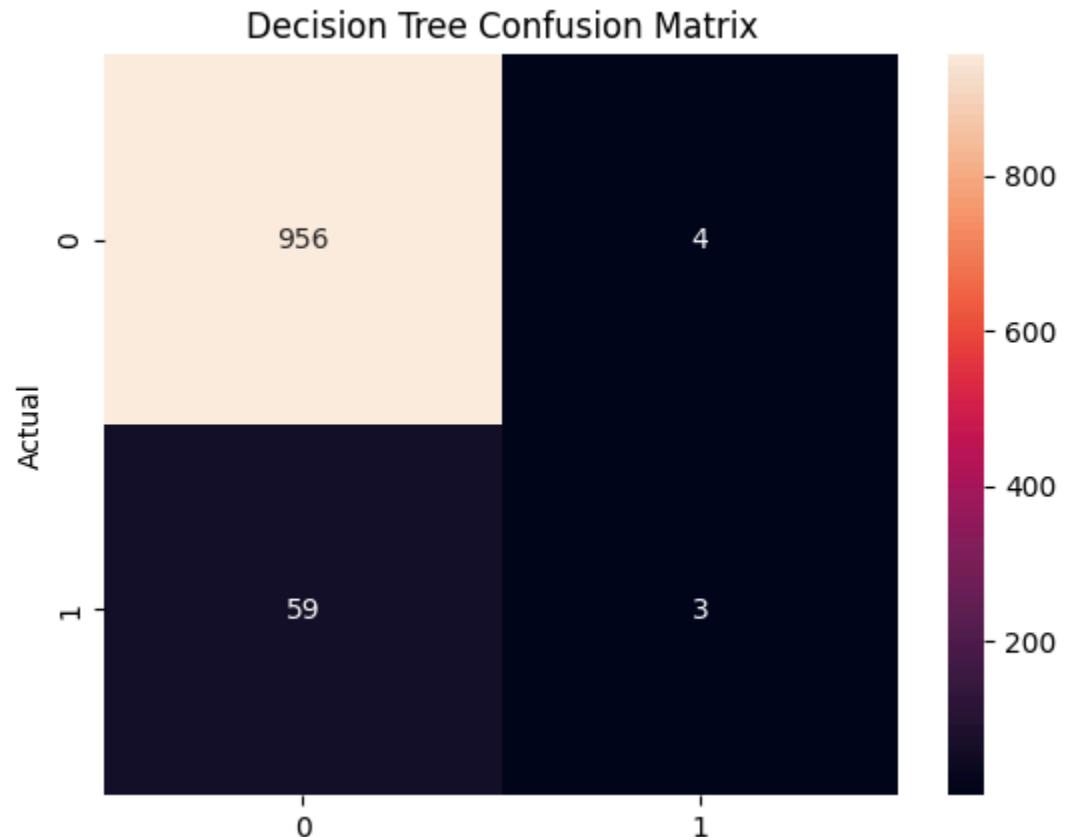
```

[[956  4]
 [ 59  3]]

```

```
Classification Report:  
precision    recall   f1-score   support  
  
          0       0.94      1.00      0.97      960  
          1       0.43      0.05      0.09       62  
  
   accuracy           0.94      1022  
macro avg       0.69      0.52      0.53      1022  
weighted avg     0.91      0.94      0.91      1022
```

/tmp/ipython-input-4207125531.py:34: FutureWarning: A value is trying to be set on a copy of a pandas Series, while the existing data for this series is not a pandas Series. A future version of pandas will change this behavior to raise an error.  
The behavior will change in pandas 3.0. This inplace method will never work because the :  
For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col}:  
df["bmi"].fillna(df["bmi"].mean(), inplace=True)



===== RANDOM FOREST RESULTS =====

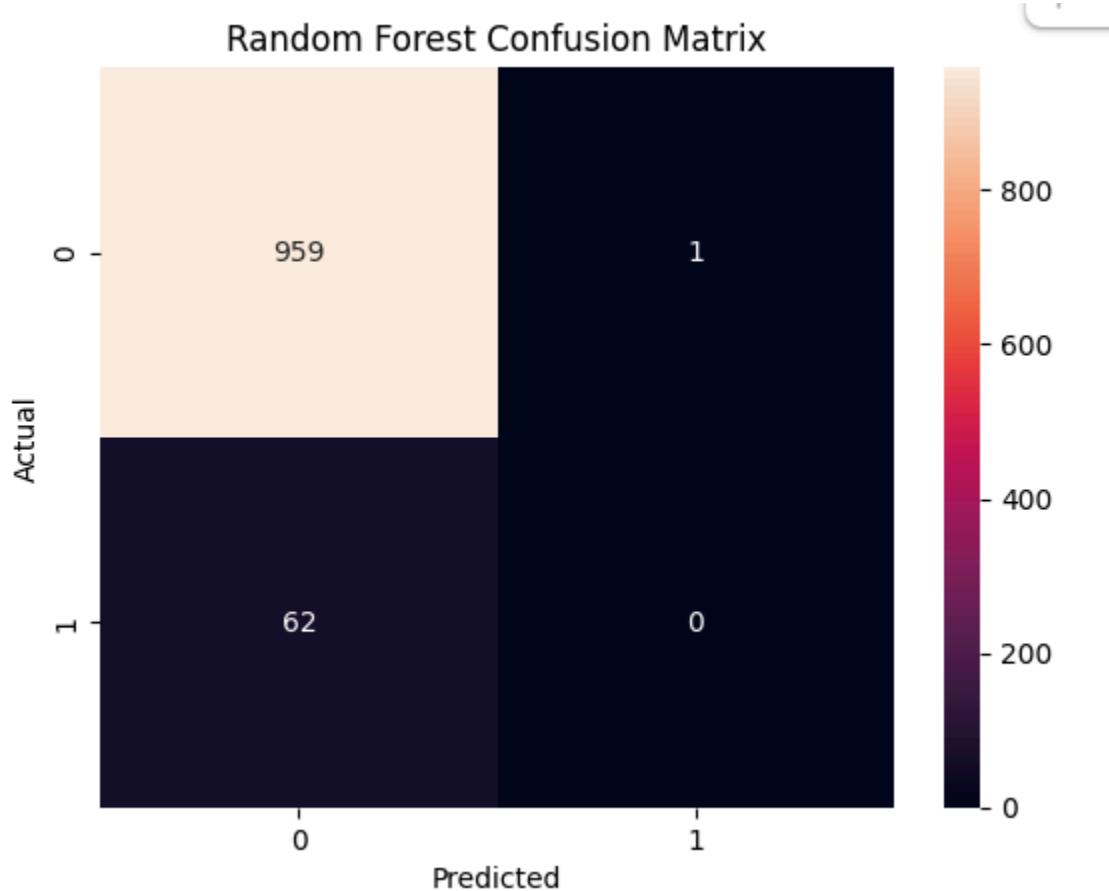
Accuracy: 0.9383561643835616

Confusion Matrix:

```
[[959  1]
 [ 62  0]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.94	1.00	0.97	960
1	0.00	0.00	0.00	62
accuracy			0.94	1022
macro avg	0.47	0.50	0.48	1022
weighted avg	0.88	0.94	0.91	1022



## Model Accuracy Comparison

↑ ↓ ⚙

