```python
import numpy as np
import pandas as pd
def f(x): return np.sin(5*x) + np.cos(2*x)
def bisec2(f,a,b):
  xi = (a + b)/2
  if f(a)*f(b) > 0: return [xi, a, b, "Root Not Bracketed"]
  elif f(xi) == 0: return [xi, a, b, "Root Found"]
  elif f(xi)*f(a) < 0: return [xi, a, xi, "Root between a and xi"]
  elif f(xi)*f(b) < 0: return [xi, xi, b, "Root between xi and b"]


def bisection(ErrorTable,ErrorNoteTable,atable,btable,xtable):
  MaxIter = 20
  eps = 0.0005
  Stopcode = "NoStopping"
  i = 0
  while i <= MaxIter and Stopcode == "NoStopping":
    ri = bisec2(f, atable[i], btable[i])
    if ri[3] == "Root Not Bracketed":
      xtable.append(ri[0])
      ErrorNoteTable.append(ri[3])
      break
    if ri[3] == "Root Found":
      xtable.append(ri[0])
      ErrorTable.append(0)
      ErrorNoteTable.append(ri[3])
      break
    atable.append(ri[1])
    btable.append(ri[2])
    xtable.append(ri[0])
    ErrorNoteTable.append(ri[3])
    if i != 0:
      ei = (xtable[i] - xtable[i - 1])/xtable[i]
      ErrorTable.append(ei)
    if abs(ErrorTable[i]) > eps:
      Stopcode = "NoStopping"
    else:
      Stopcode = "Stop"
    i+=1
  T2 = ([[i, atable[i], btable[i], xtable[i], f(atable[i]),
        f(btable[i]), f(xtable[i]), ErrorTable[i],
        ErrorNoteTable[i]] for i in range(len(xtable))])
  T2 = pd.DataFrame(T2, columns=["Iteration", "a", "b", "x_i", "f[a]", "f[b]", "f[x_i]", "er",
"ErrorNote"])
  display(T2)

# First Root
ErrorTable = [1]
ErrorNoteTable = []
atable = [-0.6]
btable = [-0.5]
```

```python
xtable = []
print("First Root")
bisection(ErrorTable,ErrorNoteTable,atable,btable,xtable)


# Second Root
ErrorTable = [1]
ErrorNoteTable = []
atable = [-0.3]
btable = [-0.2]
xtable = []
print("Second Root")
bisection(ErrorTable,ErrorNoteTable,atable,btable,xtable)


# Third Root
ErrorTable = [1]
ErrorNoteTable = []
atable = [0.1]
btable = [0.9]
xtable = []
print("Third Root")
bisection(ErrorTable,ErrorNoteTable,atable,btable,xtable)
```