Minsweeper prototype

# Pseudocode

Def drawgrid(table): # The grid will need to be drawn many times so a procedure is useful

For row in table:

For item in row:

Print(item)

Print()

#Define the starting grid as a list

Grid=[ [ “~”,“~”,“~”,“~”,“~”],

[ “~”,“~”,“~”,“~”,“~”],

[ “~”,“~”,“~”,“~”,“~”],

[ “~”,“~”,“~”,“~”,“~”],

[ “~”,“~”,“~”,“~”,“~”]]

#create the three mines

Mines=[]

Minenumber=0

While minenumber not equal to 3:

X=randomnumber as integer (0-4)

Y=randomnumber as integer (0-4)

Position=[y,x]

If position not in Mines:

Mines.append(position)

Minenumber=Minenumber+1

Minesfound=0

Guesses=0

#User enters guess

While Minesfound not equal to 3:#repeat until all 3 mines found

Validx=False

While valid x=False:#validate the x coordinate

x=input(enter a value for the x coordinate (1-5))

if x<0 or x>5:

print(“That is not a valid value”)

else:

validx=True

While valid y=False:#validate the y coordinate

y=input(enter a value for the x coordinate (1-5))

if y<0 or y>5:

print(“That is not a valid value”)

else:

validy=True

position=[y-1,x-1]#update the coordinate for the lsit

if position in Mines:#check if position in mine list

print(“You found a mine”)

Minesfound=Minesfound+1

Guesses=Guesses+1

Grid[y-1,x-1]=”M” #update the grid

Else:

Print(“You missed the mines”)

Guesses=Guesses+1

Grid[y-1,x-1]=”X” #update the grid

Drawgrid(grid)

Print(“Congratulations! You found all the mines!”)

Print(“You took”, guesses,”guesses”)

# Exercise:

Use this code to create the solution. Remember that you must show the development of this final code.