Elizabeth Perez

Professor Monshi

CMSC 203

3/27/2020

UML Class Diagram + Psuedo-Code

|  |
| --- |
| Property.java |
| propertyName : String  city : String  rentAmount : double  owner : String  plot : Plot |
| Property:  Get a Property called property  Set propertyName to property’s propertyName  Set city to property’s city  Set rentAmount to property’s rentAmount  Set owner to property’s owner  Set plot to property’s plot  Property :  Get a String called name, a String called location, a double called rentalFee, a String called propertyOwner, an int x, an int y, an int depth, and an int width as parameters  Set propertyName to name  Set city to location  Set rentAmount to rentalFee  Set owner to propertyOwner  Set plot to a Plot at position (x,y), depth of depth, and width of width  Property :  Get a String called name, a String called location, a double called rentalFee, and a String called propertyOwner as parameters  Set propertyName to name  Set city to location  Set rentAmount to rentalFee  Set owner to propertyOwner  toString :  Declare a String called message  Set message to “Property Name: ” + name + “\nLocated in ” + city + “\nBelonging to ” + owner + “\nRent Amount: ” + rentAmount  Return message  GetName :  Return name  GetCity :  Return city  GetRentalAmount :  Return rentAmount  GetOwner :  Return owner  GetPlot :  Return plot  SetName :  Get a string called propertyName as a parameter  Set name to propertyName  SetCity :  Get a string called location as a parameter  Set city to location  SetRentalAmount :  Get a double called rentalFee as a parameter  Set rentalAmount to rentalFee  SetOwner :  Get a string called propertyOwner as a parameter  Set owner to propertyOwner  SetPlot :  Get an int called x, an int called y, an int called depth, and an int called width as parameters  Set plot to a Plot at position (x,y), depth of depth, and width of width |

|  |
| --- |
| Plot.java |
| x : int  y : int  depth : int width : int |
| Plot:  Get a Plot called plot as a parameter  Set x to plot’s x  Set y to plot’s y  Set width to plot’s width  Set depth to plot’s depth  Plot :  Get an int called xPosition, an int called yPosition, an int called propertyDepth, and an int called propertyWidth as parameters  Set x to xPosition  Set y to yPosition  Set depth to propertyDepth  Set width to propertyWidth  ToString :  Declare a String called message  Set message to "Upper left: (" + x + "," + y + "); Width: " + width + " Depth: " + depth  Return message  Overlaps :  Get a Plot called plot as a parameter  Declare an int called x2 and initialize it to plot’s x value  Declare an int called y2 and initialize it to plot’s y value  Declare an int called depth2 and initialize it to plot’s depth value  Declare an int called width2 and initialize it to plot’s width value  Declare a boolean called overlap and initialize it to false  If (x2 is between x and x+width)  If (y2 is between y-depth and y) or (y2-depth2 is between y-depth and y)  Set overlap to true  If (x2+width2 is between x and x+width)  If (y2 is between y-depth and y) or (y2-depth2 is between y-depth and y)  Set overlap to true  If (x is between x2 and x2+width2)  If (y is between y2-depth2 and y2) or (y-depth is between y2-depth2 and y2)  Set overlap to true  If (x+width is between x2 and x2+width2)  If (y is between y2-depth2 and y2) or (y-depth is between y2-depth2 and y2)  Set overlap to true  If this plot encompasses plot  Set overlap to true  Return overlap  Encompasses :  Get a Plot called plot as a parameter  Declare an int called x2 and initialize it to plot’s x value  Declare an int called y2 and initialize it to plot’s y value  Declare an int called depth2 and initialize it to plot’s depth value  Declare an int called width2 and initialize it to plot’s width value  If (x2 is between x and x+width)  If (y2 is between y-depth and y)  If (x2+width2 is between x and x+width)  If (y2-depth2 is between y-depth and y)  Return true  Return false  getX :  Return x  getY :  Return y  getDepth :  Return depth  getWidth :  Return width  setX:  Get an int xPosition as a parameter  Set x to xPosition  setY:  Get an int yPosition as a parameter  Set y to yPosition  setWidth:  Get an int propertyWidth as a parameter  Set width to propertyWidth  setDepth:  Get an int propertyDepth as a parameter  Set depth to propertyDepth |

|  |
| --- |
| ManagementCompany.java |
| name : String  taxID : String  mgmFeePer : double  MAX\_PROPERTY : int : 5  properties : Property[MAX\_PROPERTY]  count : int  MGMT\_WITDH : int : 10  MGMT\_DEPTH : int : 10  plot : Plot |
| ManagementCompany:  Set name to “”  Set taxID to “”  Set mgmFeePer to 0  Set plot to a Plot at with x value at 0, y value at 0, width value at MGMT\_WIDTH, and depth value at MGMT\_DEPTH  Set properties to a Property array with a length of MAX\_PROPERTY  Set count to 0  ManamementCompany :  Get a String called companyName, a String called id, and a double called fee as parameters  Set name to companyName  Set taxID to id  Set mgmFeePer to fee  Set plot to a Plot at with x value at 0, y value at 0, width value at MGMT\_WIDTH, and depth value at MGMT\_DEPTH  Set properties to a Property array with a length of MAX\_PROPERTY  Set count to 0  ManamementCompany :  Get a String called companyName, a String called id, and a double called fee, an int called x, an int called y, an int called depth, and an int called width as parameters  Set name to companyName  Set taxID to id  Set mgmtFeePer to fee  Set plot to a Plot at with x value at x, y value at y, width value at width, and depth value at depth  Set properties to a Property array with a length of MAX\_PROPERTY  Set count to 0  ManagementCompany:  Get a ManagementCompany called company as a parameter  Set name to company’s name  Set taxID to company’s taxID  Set mgmFeePer to company’s mgmFeePer  Set plot to company’s plot  Set properties to company’s properties  Set count to company’s count  toString :  Declare a String called message  Set message to “List of the properties for ” + name + “, taxID: ” + taxID  Add "\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n" to message  For an int called i set to 0, i less than count, i plus 1  Add on properties at index i toString to message  Add “\n” to message  Add "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n" to message  Declare a double called totalMGMFee and set to (totalRent \* mgmFeePer) / 100  Add on “Total Management Fee: ” + t totalMGMFee to message  Return message  addProperty :  Get a Property called property as a parameter  If count is equal to MAX\_PROPERTY  Return –1  If propery is null  Return –2  If property’s plot is not encompassed by plot  Return –3  For an int called i set to 0, i less than count, i plus 1  If property’s plot overlaps properties at index i’s plot  Return -4  Declare a Property called prop and set its values to the same as property  Set properties at index count to prop  Add 1 to count  Return count minus 1  addProperty :  Get a String called propertyName, a String called city, a double called rent and a String called ownerName as parameters  If count is equal to MAX\_PROPERTY  Return –1  Declare a Property called prop and set to a Property with name value as propertyName, city value as city, rentalAmount as rent, and owner as ownerName  Set properties at index count to prop  Add 1 to count  Return count minus 1  addProperty :  Get a String called propertyName, a String called city, a double called rent and a String called ownerName, an int called x, an int called y, an int called width and an int called depth as parameters  If count is equal to MAX\_PROPERTY  Return –1  Declare a Property called prop and set to a Property with name value as propertyName, city value as city, rentalAmount as rent, owner as ownerName, x value as x, y value as y, width value as width, and depth value as depth  If prop’s plot is not encompassed by plot  Return –3  For an int called i set to 0, i less than count, i plus 1  If prop’s plot overlaps properties at index i’s plot  Return -4  Set properties at index count to prop  Add 1 to count  Return count minus 1  displayPropertyAtIndex :  Det an int called i as a parameter  Return properties at index i's toString  getMAX\_PROPERTY :  Return MAX\_PROPERTY  totalRent :  Declare a double called total and initialize to 0  For an int called i set to 0, i less than count, i plus 1  Add properties at index i’s rentalAmount to total  Return total  maxRentPropertyIndex :  Declare an int called index and initialize to 0  Declare a double called highest and initialize to properties at index 0’s rentalAmount  For an int called i set to 1, i less than count, i plus 1  If properties at index i's rentalAmount is greater than highest  Set highest to properties at index i's rentalAmount  Set index to i  Return index  maxRentProp :  Declare a double called max and set to properties at index maxRentPropertyIndex’s rentalAmount  Return max |