Lab: Interfaces and Abstraction

1. Shapes

Build hierarchy of interfaces and classes:

|  |
| --- |
| Rectangle |
| -width: Integer  -height: Integer |

|  |
| --- |
| <<inteface>>  <<Drawable>> |
| +Draw() |

|  |
| --- |
| Circle |
| -radius: Integer |

You should be able to use the class like this:

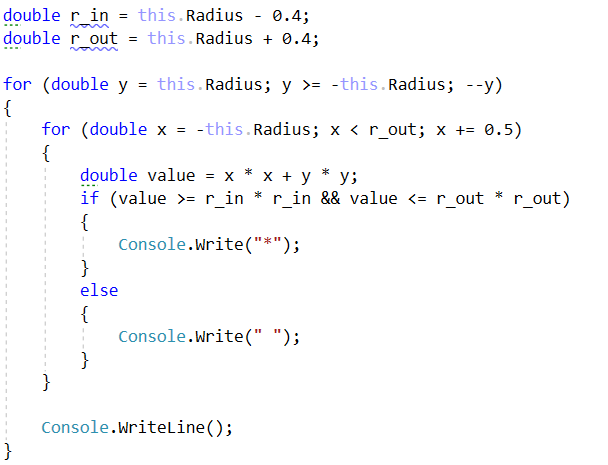
|  |
| --- |
| StartUp.cs |
| var radius = int.Parse(Console.ReadLine());  IDrawable circle = new Circle(radius);  var width = int.Parse(Console.ReadLine());  var height = int.Parse(Console.ReadLine());  IDrawable rect = new Rectangle(width, height);  circle.Draw();  rect.Draw(); |

Examples

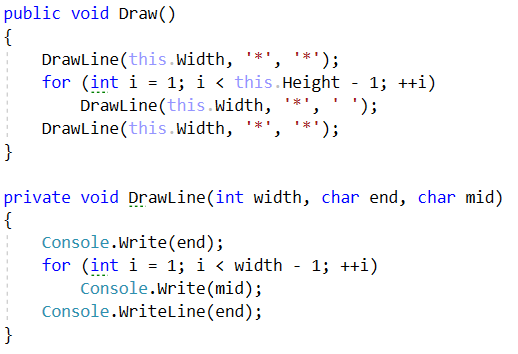
|  |  |
| --- | --- |
| Input | Output |
| 3  5  4 | \*\*\*\*\*\*\*  \*\* \*\*  \*\* \*\*  \* \*  \*\* \*\*  \*\* \*\*  \*\*\*\*\*\*\*  \*\*\*\*  \* \*  \* \*  \* \*  \*\*\*\* |

Solution

For circle drawing you can use this algorithm:



For rectangle drawing algorithm will be:



1. Cars

Build hierarchy of interfaces and classes:

|  |
| --- |
| <<ICar>> |
| +Model: string  +Color: string  +Start(): string  +Stop(): string |

|  |
| --- |
| <<IElectricCar>> |
| +Battery: int |

|  |
| --- |
| Tesla |
| +ToString(): string |

|  |
| --- |
| Seat |
| +ToString(): string |

Your hierarchy have to be used with this code

|  |
| --- |
| Main.java |
| ICar seat = new Seat("Leon", "Grey");  ICar tesla = new Tesla("Model 3", "Red", 2);  Console.WriteLine(seat.ToString());  Console.WriteLine(tesla.ToString()); |

Examples

|  |  |
| --- | --- |
| Input | Output |
|  | Grey Seat Leon  Engine start  Breaaak!  Red Tesla Model 3 with 2 Batteries Engine start  Breaaak! |