EXAMPLE 28: FUNCTIONS THAT RETURN FUNCTIONS

EXAMPLE 28: FUNCTIONS THAT RETURN FUNCTIONS

SAY WE HAVE A RECTANGLE OBJECT AND A CIRCLE OBJECT

SAY WE HAVE A RECTANGLE OBJECT AND A CIRCLE OBJECT

AND GIVEN AN OBJECT (NOT SURE WHETHER CIRCLE OR RECTANGLE), WE NEEDED A WAY TO FIND ITS AREA

SAY WE HAVE A RECTANGLE OBJECT AND A CIRCLE OBJECT

AND GIVEN AN OBJECT (NOT SURE WHETHER CIRCLE OR RECTANGLE), WE NEEDED A WAY TO FIND ITS AREA

SAY WE HAVE A RECTANGLE OBJECT AND A CIRCLE OBJECT

AND GIVEN AN OBJECT (NOT SURE WHETHER CIRCLE OR RECTANGLE), WE NEEDED A WAY TO FIND ITS AREA

SAY WE HAVE A RECTANGLE OBJECT AND A CIRCLE OBJECT

AND GIVEN AN OBJECT (NOT SURE WHETHER CIRCLE OR RECTANGLE), WE NEEDED A WAY TO FIND ITS AREA

EASY PEASY! JUST NEED A FUNCTION THAT RETURNS THE CORRECT AREA FUNCTION!

A FUNCTION THAT RETURNS THE CORRECT AREA FUNCTION!

```
function getAreaFunction(shape) {
  var circleAreaFunction = function(s) {
     return 3.14 * s.radius * s.radius;
  }
  var rectangleAreaFunction = function(s) {
      return s.length * s.breadth;
  }

  if (shape instanceof Rectangle) {
    console.log("Rectangle was passed in");
    return rectangleAreaFunction;
  }
  if (shape instanceof Circle) {
    console.log("Circle was passed in");
    return circleAreaFunction;
  }
  return undefined;
}
```

A FUNCTION

```
function getAreaFunction(shape) {
  var circleAreaFunction = function(s) {
     return 3.14 * s.radius * s.radius;
  }
  var rectangleAreaFunction = function(s) {
      return s.length * s.breadth;
  }

  if (shape instanceof Rectangle) {
    console.log("Rectangle was passed in");
    return rectangleAreaFunction;
  }
  if (shape instanceof Circle) {
    console.log("Circle was passed in");
    return circleAreaFunction;
  }
  return undefined;
}
```

A FUNCTION

```
function getAreaFunction(shape) {
  var circleAreaFunction = function(s) {
     return 3.14 * s.radius * s.radius;
  }
  var rectangleAreaFunction = function(s) {
      return s.length * s.breadth;
  }

  if (shape instanceof Rectangle) {
    console.log("Rectangle was passed in");
    return rectangleAreaFunction;
  }
  if (shape instanceof Circle) {
    console.log("Circle was passed in");
    return circleAreaFunction;
  }
  return undefined;
}
```

A FUNCTION

```
function getAreaFunction(shape) {
  var circleAreaFunction = function(s) {
     return 3.14 * s.radius * s.radius;
  }
  var rectangleAreaFunction = function(s) {
      return s.length * s.breadth;
  }

  if (shape instanceof Rectangle) {
    console.log("Rectangle was passed in");
    return rectangleAreaFunction;
  }
  if (shape instanceof Circle) {
    console.log("Circle was passed in");
    return circleAreaFunction;
  }
  return undefined;
}
```

CREATE A VARIABLE HOLDING A FUNCTION THAT FINDS THE AREA OF A CIRCLE

A FUNCTION

```
function getAreaFunction(shape) {
  var circleAreaFunction = function(s) {
     return 3.14 * s.radius * s.radius;
  }
  var rectangleAreaFunction = function(s) {
     return s.length * s.breadth;
  }
  if (shape instanceof Rectangle) {
    console.log("Rectangle was passed in");
    return rectangleAreaFunction;
  }
  if (shape instanceof Circle) {
    console.log("Circle was passed in");
    return circleAreaFunction;
  }
  return undefined;
}
```

CREATE ANOTHER VARIABLE HOLDING A FUNCTION THAT FINDS THE AREA OF A RECTANGLE

A FUNCTION

USE instanceof TO TEST IF THIS IS A RECTANGLE

```
function getAreaFunction(shape) {
  var circleAreaFunction = function(s) {
     return 3.14 * s.radius * s.radius;
  }
  var rectangleAreaFunction = function(s) {
      return s.length * s.breadth;
  }
  if (shape instanceof Rectangle) {
    console.log("Rectangle was passed in");
    return rectangleAreaFunction;
  }
  if (shape instanceof Circle) {
    console.log("Circle was passed in");
    return circleAreaFunction;
  }
  return undefined;
}
```

A FUNCTION

IF YES, RETURN THE RECTANGLE AREA FUNCTION!

```
function getAreaFunction(shape) {
  var circleAreaFunction = function(s) {
     return 3.14 * s.radius * s.radius;
  }
  var rectangleAreaFunction = function(s) {
      return s.length * s.breadth;
  }
  if (shape instanceof Rectangle) {
    console.log("Rectangle was passed in");
    return rectangleAreaFunction;
  }
  if (shape instanceof Circle) {
    console.log("Circle was passed in");
    return circleAreaFunction;
  }
  return undefined;
}
```

USE instanceof TO TEST IF THIS IS A CIRCLE

A FUNCTION

```
function getAreaFunction(shape) {
  var circleAreaFunction = function(s) {
     return 3.14 * s.radius * s.radius;
  }
  var rectangleAreaFunction = function(s) {
      return s.length * s.breadth;
  }
  if (shape instanceof Rectangle) {
    console.log("Rectangle was passed in");
    return rectangleAreaFunction;
  }
  if (shape instanceof Circle) {
    console.log("Circle was passed in");
    return circleAreaFunction;
  }
  return undefined;
}
```

A FUNCTION

IF YES, RETURN THE CIRCLE AREA FUNCTION!

```
function getAreaFunction(shape) {
  var circleAreaFunction = function(s) {
     return 3.14 * s.radius * s.radius;
  }
  var rectangleAreaFunction = function(s) {
      return s.length * s.breadth;
  }

  if (shape instanceof Rectangle) {
    console.log("Rectangle was passed in");
    return rectangleAreaFunction;
  }
  if (shape instanceof Circle) {
    console.log("Circle was passed in");
    return circleAreaFunction;
  }
  return undefined;
}
```

A FUNCTION

```
function getAreaFunction(shape) {
  var circleAreaFunction = function(s) {
     return 3.14 * s.radius * s.radius;
  }
  var rectangleAreaFunction = function(s) {
      return s.length * s.breadth;
  }

  if (shape instanceof Rectangle) {
    console.log("Rectangle was passed in");
    return rectangleAreaFunction;
  }
  if (shape instanceof Circle) {
    console.log("Circle was passed in");
    return circleAreaFunction;
  }
  return undefined;
}
```

IF NEITHER VARIETY, THEN RETURN UNDEFINED.

SAY WE HAVE A RECTANGLE OBJECT AND A CIRCLE OBJECT

AND GIVEN AN OBJECT (NOT SURE WHETHER CIRCLE OR RECTANGLE), WE NEEDED A WAY TO FIND ITS AREA

EASY PEASY! JUST NEED A FUNCTION THAT RETURNS THE CORRECT AREA FUNCTION!

SAY WE HAVE A RECTANGLE OBJECT AND A CIRCLE OBJECT

AND GIVEN AN OBJECT (NOT SURE WHETHER CIRCLE OR RECTANGLE), WE NEEDED A WAY TO FIND ITS AREA

EASY PEASY! JUST NEED A FUNCTION THAT RETURNS THE CORRECT AREA FUNCTION!

AND GIVEN AN OBJECT (NOT SURE WHETHER CIRCLE OR RECTANGLE), WE NEEDED A WAY TO FIND ITS AREA

```
function objectStuff() {
  var rectangle1 = new Rectangle(5, 5,"Blue");
  var circle2 = new  Circle(10,"Red");
  var areaFunction1 = getAreaFunction(rectangle1);
  var areaFunction2 = getAreaFunction(circle2);


  console.log("Area of circle is : " +
areaFunction2(circle2));
  console.log("Area of rectangle  is : " +
areaFunction1(rectangle1));

}
```

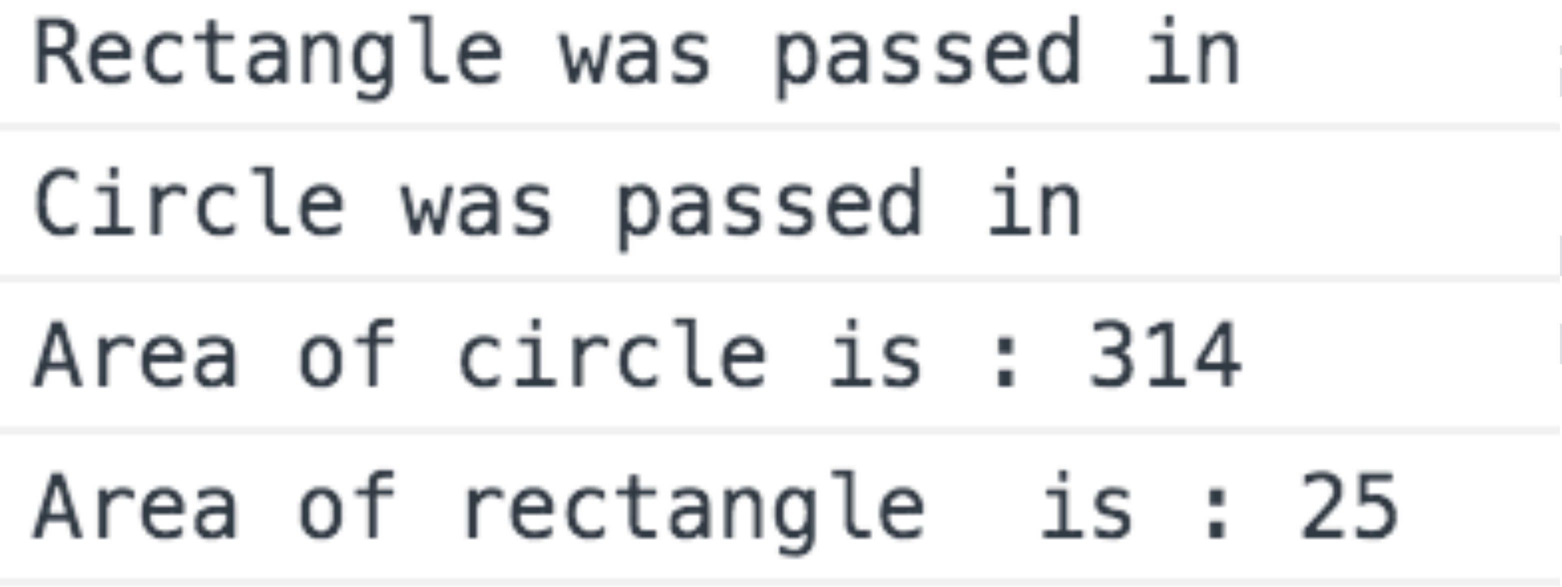
THIS IS NOW QUITE EASY TO USE.

AND GIVEN AN OBJECT (NOT SURE WHETHER CIRCLE OR RECTANGLE), WE NEEDED A WAY TO FIND ITS AREA

```
function objectStuff() {
  var rectangle1 = new Rectangle(5, 5,"Blue");
  var circle2 = new  Circle(10,"Red");
  var areaFunction1 = getAreaFunction(rectangle1);
  var areaFunction2 = getAreaFunction(circle2);


  console.log("Area of circle is : " +
areaFunction2(circle2));
  console.log("Area of rectangle  is : " +
areaFunction1(rectangle1));

}
```

THIS IS NOW QUITE EASY TO USE.

```
function objectStuff() {
                                                  ,"Blue");
                                                  ;
                                                  (rectangle1);
                                                  (circle2);

  console.log("Area of circle is : " +
areaFunction2(circle2));
  console.log("Area of rectangle  is : " +
areaFunction1(rectangle1));

}
```



```
Rectangle was passed in
Circle was passed in
Area of circle is : 314
Area of rectangle  is : 25
```