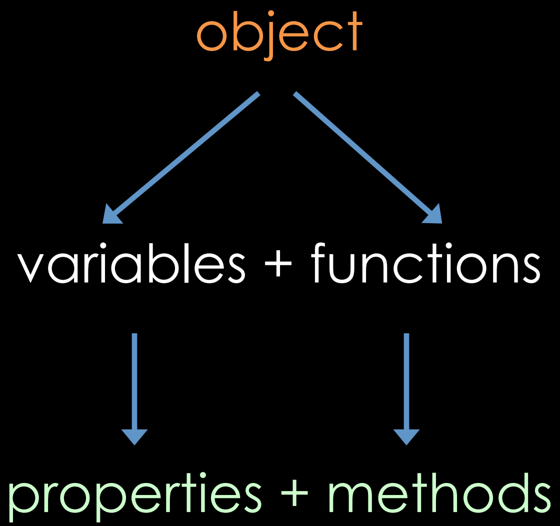
# Object

A group of variables and functions ex. hotel object



object

variables + functions

properties + methods

# Object

Properties describe the objects (ex. name, # of rooms, etc.)

Methods describe tasks associated with the object (ex. check availability of hotel rooms)

# Objects

Browser

object model

JavaScript

Global objects

Document

Object model

(DOM)

Representation

of the current

wen page

School of

Design

# Global JavaScript objects

| | |
|---|---|
| String | for working with string values |
| Math | for working with numbers and calculations |
| Date | to represent and handle dates |

| | |
|---|---|
| Math.PI(); | Math’s object PI property will return the value of PI |

# Date object

Date to represent and handle dates

```
var today = new Date();  //holds today's date and the current time
```

# Date object - methods

| | |
|---|---|
| getDate() | Get the day as a number (1-31) |
| getDay() | Get the weekday a number (0-6) |
| getFullYear() | Get the four digit year (yyyy) |
| getHours() | Get the hour (0-23) |
| getMilliseconds() | Get the milliseconds (0-999) |
| getMinutes() | Get the minutes (0-59) |
| getMonth() | Get the month (0-11) |
| getSeconds() | Get the seconds (0-59) |
| getTime() | Get the time (milliseconds since January 1, 1970) |

# Date object - methods

| | |
|---|---|
| getTimezoneOffset() | Returns time zone offset in mins for locale |
| toDateString() | Returns “date” as a human-readable string |
| toTimeString() | Returns “time” as a human-readable string |
| toString() | Returns a string representing a specified date |

# Date object - methods

|  |  |
|---|---|
| toDateString() | Returns “date” as a human-readable string |

Mon Sept 15 2014.

# Date object - methods

```
// Create a variable to hold a new Date object (defaults to now)
 var today = new Date();
// Create a variable to hold the year this year
var year = today.getFullYear();
// Create a variable called el to hold the element whose id
attribute //has a value of footer
var el = document.getElementById('footer');
// Write the year into that element.
el.innerHTML = '<p>Copyright &copy;' + year + '</p>';
```

# Date object - methods

```
<!DOCTYPE html>
<html>
  <head>
    <title>Objects - Date Object</title>
    <link rel="stylesheet" href="css/c03.css" />
  </head>
  <body>
    <h1>TravelWorthy</h1>
    <div id="footer"></div>
    <script src="js/date-object.js"></script>
  </body>
</html>
```

# Date object - methods

Exercise – add copyright note to your website using Date object

# Date object - methods

```
// Create a variable to hold a new Date object (defaults to now)
var today = new Date();
// Get the year this year
var year = today.getFullYear();
// Set the date that the company was established
var est = new Date('Apr 16, 1996 15:45:55');
// Get difference between then & now in milliseconds
var difference = today.getTime() - est.getTime();
// Divide by number of milliseconds to get years
difference = (difference / 31556900000);
```

# Date object - methods

```
// Create a variable called elMsg to hold the element whose id
//attribute has a value of message
var elMsg = document.getElementById('message');

// Write the message into that element.
elMsg.textContent = Math.floor(difference) + ' years of online travel
advice';
```

# Date object - methods

```
<!DOCTYPE html>
<html>
  <head>
    <title>Date Object Difference</title>
    <link rel="stylesheet" href="css/c03.css" />
  </head>
  <body>
    <h1>TravelWorthy</h1>
    <div id="message">Established 1945</div>
    <script src="js/date-object-difference.js"></script>
  </body>
</html>
```

# Date object - methods

Exercise – write a function to return your friend's age

# Objects

Functions are set of related statements together that represent a single task

Functions can take parameters (information required to do their job) and may return a value

An object is a series of variables and functions grouped together that represent something from the world around you

In an object, variables are known as properties and functions as methods of the object

# Objects

Web browsers implement objects that represent both the browser window and the document loaded into the browser window

JavaScript has several built-in objects such as String, Math, Number and Date.

Arrays and objects can be used to create complex data sets

# Objects – example.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Objects - Example</title>
    <link rel="stylesheet" href="css/c03.css" />
  </head>
  <body>
    <h1>TravelWorthy</h1>
    <div id="info">
      <h2>latest hotel offer</h2>
```

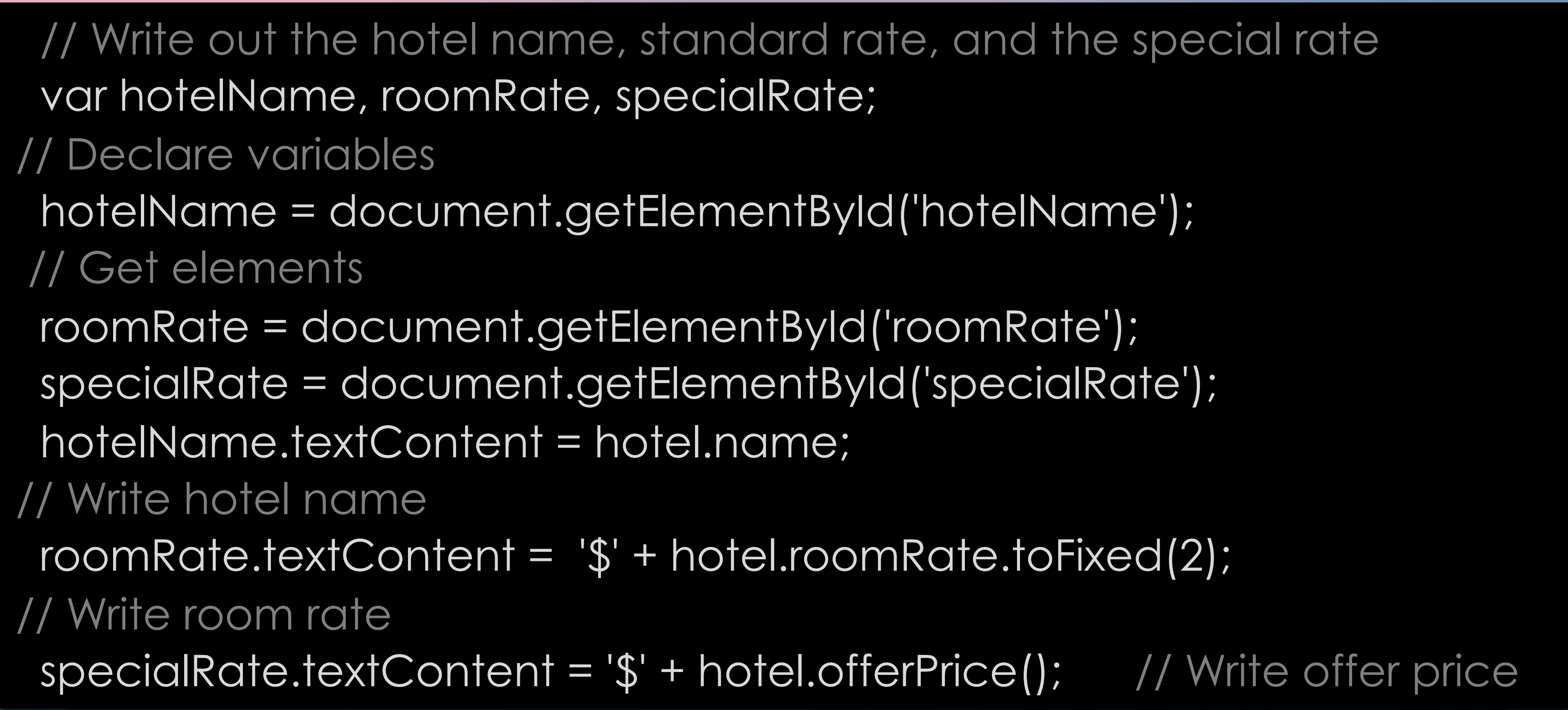
# Objects – example.html

```
<div id="hotelName"></div>
      <div id="roomRate"></div>
      <div id="specialRate"></div>
      <p>room rate when you book 2 or more nights</p>
      <div id="offerEnds"></div>
    </div>
    <script src="js/example.js"></script>
  </body>
</html>
```

# Objects – example.html

```
(function() {
 // PART ONE: CREATE HOTEL OBJECT AND WRITE OUT THE OFFER DETAILS
 // Create a hotel object using object literal syntax
 var hotel = {
   name: 'Park',
   roomRate: 240, // Amount in dollars
   discount: 15,  // Percentage discount
   offerPrice: function() {
     var offerRate = this.roomRate * ((100 - this.discount) / 100);
     return offerRate;
   }
 }
```

# Objects – example.html



```
 // Write out the hotel name, standard rate, and the special rate
 var hotelName, roomRate, specialRate;
// Declare variables
 hotelName = document.getElementById('hotelName');
 // Get elements
 roomRate = document.getElementById('roomRate');
 specialRate = document.getElementById('specialRate');
 hotelName.textContent = hotel.name;
// Write hotel name
 roomRate.textContent =  '$' + hotel.roomRate.toFixed(2);
// Write room rate
 specialRate.textContent = '$' + hotel.offerPrice();     // Write offer price
```

# Objects – example.html

```
// PART TWO: CALCULATE AND WRITE OUT THE EXPIRY DETAILS FOR THE OFFER
var expiryMsg; // Message displayed to users
var today;     // Today's date
var elEnds;    // The element that shows the message about the offer ending
function offerExpires(today) {
  // Declare variables within the function for local scope
  var weekFromToday, day, date, month, year, dayNames, monthNames;
  // Add 7 days time (added in milliseconds)
  weekFromToday = new Date(today.getTime() + 7 * 24 * 60 * 60 * 1000);
```

# Objects – example.html

```
// Create arrays to hold the names of days / months
  dayNames = ['Sunday', 'Monday', 'Tuesday', 'Wednesday', 'Thursday',
'Friday', 'Saturday'];
  monthNames = ['January', 'February', 'March', 'April', 'May', 'June', 'July',
'August', 'September', 'October', 'November', 'December'];
  // Collect the parts of the date to show on the page
  day = dayNames[weekFromToday.getDay()];
  date = weekFromToday.getDate();
  month = monthNames[weekFromToday.getMonth()];
  year = weekFromToday.getFullYear();
  // Create the message
  expiryMsg = 'Offer expires next ';
  expiryMsg += day + ' <br />(' + date + ' ' + month + ' ' + year + ')';
  return expiryMsg;
```

# Objects – example.html

```
today = new Date();                          // Put today's date in variable
elEnds = document.getElementById('offerEnds');  // Get the offerEnds element
elEnds.innerHTML = offerExpires(today);        // Add the expiry message

// Finish the immediately invoked function expression
}());
```

# Objects – example.html

Exercise- write a script to display the restaurant schedule for next week using Date and Time object