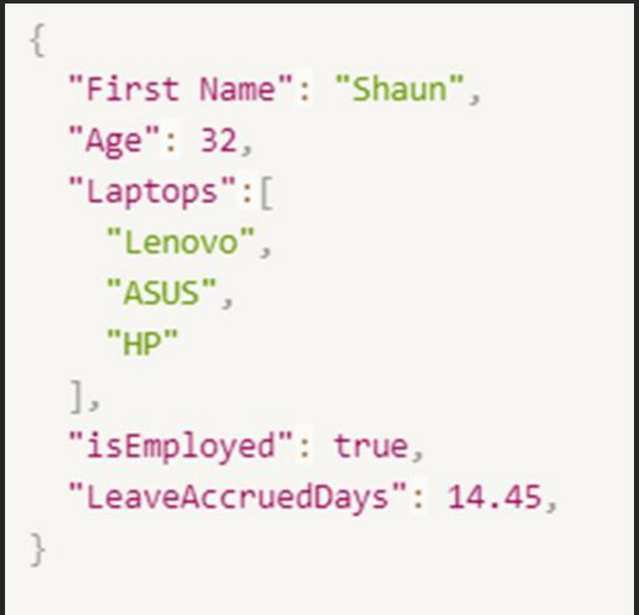
JSON Files– (Web Data Cluster)

Diploma IT - Advanced Programming

What is JSON?



```
{
  "First Name": "Shaun",
  "Age": 32,
  "Laptops":[
    "Lenovo",
    "ASUS",
    "HP"
  ],
  "isEmployed": true,
  "LeaveAccruedDays": 14.45,
}
```

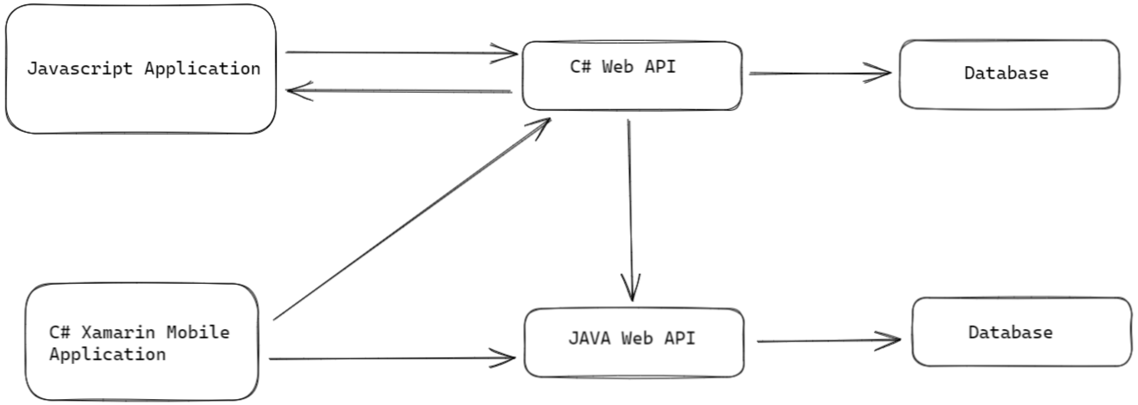
JSON stands for JavaScript Object Notation and is a way of formatting objects in a string by using various characters and structures to define the details of the object.

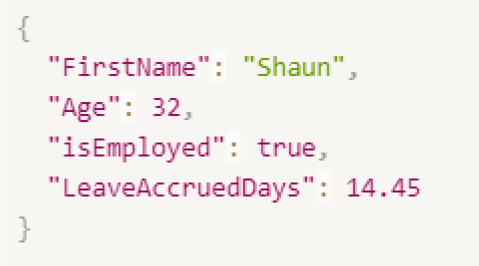
Originally developed for JavaScript, this structure is now commonly used as a data interchange format to pass data between different systems and languages in a format that can be easily interpreted by any language.

JSON can be used to represent almost any data type including arrays and nested objects and has been replacing XML as the standard transfer language of the internet.

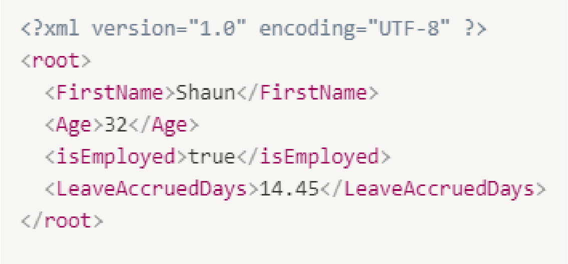
JSON Usage

- Language Agnostic Web Service communication
- Transferring data across the internet
- Payload in HTTP Requests
- Replaces XML as a data-transfer language





```json
{
  "FirstName": "Shaun",
  "Age": 32,
  "isEmployed": true,
  "LeaveAccruedDays": 14.45
}
```



```xml
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <FirstName>Shaun</FirstName>
  <Age>32</Age>
  <isEmployed>true</isEmployed>
  <LeaveAccruedDays>14.45</LeaveAccruedDays>
</root>
```

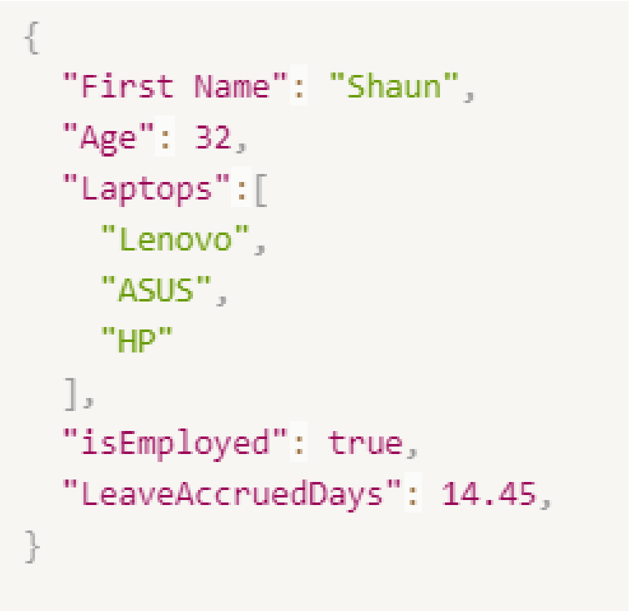
JSON Structure

All JSON objects always start and end with Curly Brackets for a single object or Square Brackets when dealing with collections

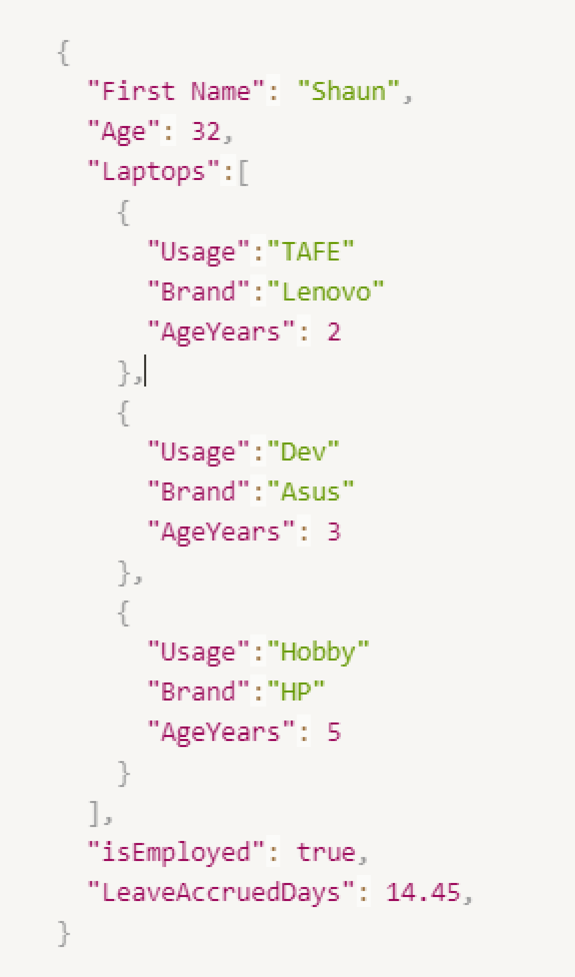
Every field in the structure is then defined by a key – value pair. If multiple fields are needed, each set is separated by commas.

The key is always enclosed in double quotes and represents the field or sub-object’s name. The key is usually camelCase

The value is placed after the key and can represent a single value, or a structure such as an array or object which holds several values.



```
{
  "First Name": "Shaun",
  "Age": 32,
  "Laptops":[
    "Lenovo",
    "ASUS",
    "HP"
  ],
  "isEmployed": true,
  "LeaveAccruedDays": 14.45,
}
```



```
{
  "First Name": "Shaun",
  "Age": 32,
  "Laptops":[
    {
      "Usage":"TAFE"
      "Brand":"Lenovo"
      "AgeYears": 2
    },
    {
      "Usage":"Dev"
      "Brand":"Asus"
      "AgeYears": 3
    },
    {
      "Usage":"Hobby"
      "Brand":"HP"
      "AgeYears": 5
    }
  ],
  "isEmployed": true,
  "LeaveAccruedDays": 14.45,
}
```

Main JSON Data Types

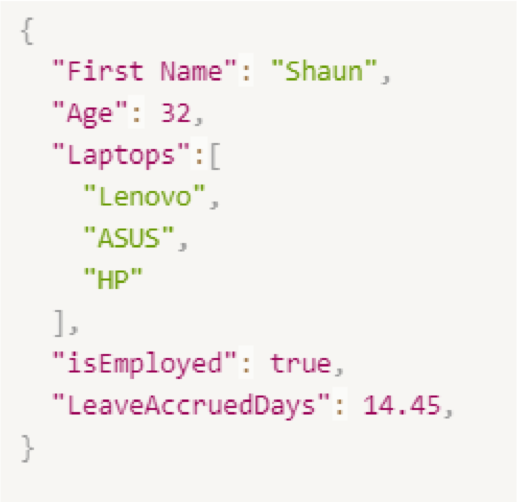
Strings – must always be surrounded by double quotes like in most programming languages.

Numbers – Can represent whole or decimal numbers, written without quotes.

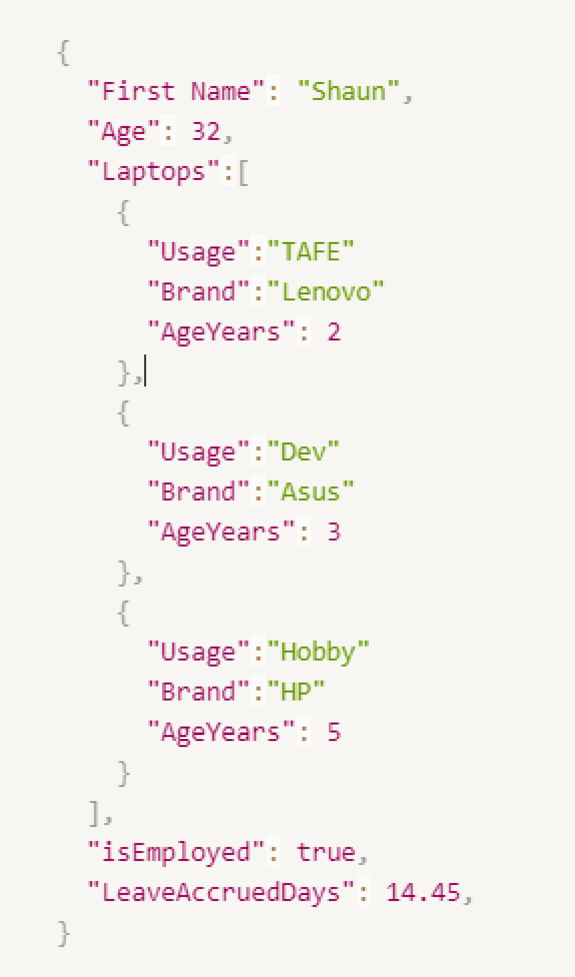
Boolean – data is written with either a true or false value

Arrays – Starts with a key, which is then followed by square brackets for its value. Within the brackets the array entries are separated by brackets.

Objects – JSON objects can contain nested objects, or an array of objects. An object is defined by adding a key followed by curly braces as its value. This set of curly braces can then hold its own key-value pairs.



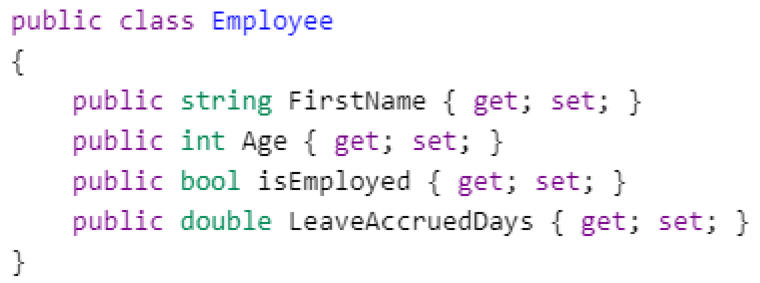
```
{
  "First Name": "Shaun",
  "Age": 32,
  "Laptops":[
    "Lenovo",
    "ASUS",
    "HP"
  ],
  "isEmployed": true,
  "LeaveAccruedDays": 14.45,
}
```



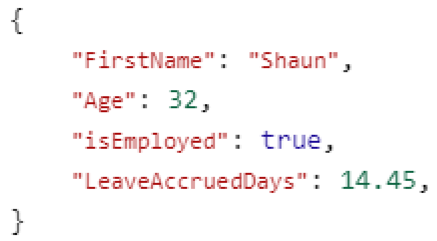
```
{
  "First Name": "Shaun",
  "Age": 32,
  "Laptops":[
    {
      "Usage":"TAFE"
      "Brand":"Lenovo"
      "AgeYears": 2
    },
    {
      "Usage":"Dev"
      "Brand":"Asus"
      "AgeYears": 3
    },
    {
      "Usage":"Hobby"
      "Brand":"HP"
      "AgeYears": 5
    }
  ],
  "isEmployed": true,
  "LeaveAccruedDays": 14.45,
}
```

JSON Conversion

- Converting a string of JSON data to C# Objects is referred to as **Deserialization**
- Converting C# Objects to a JSON String is referred to as **Serialization**
- Normally you will have a model in your code that will map to the JSON data when Serializing or Deserializing data



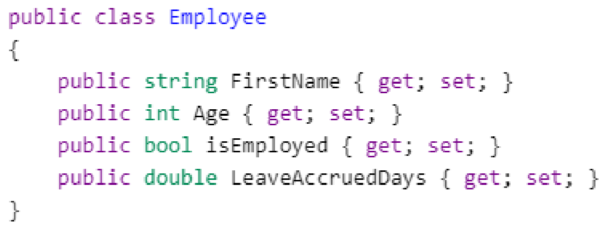
```
public class Employee
{
    public string FirstName { get; set; }
    public int Age { get; set; }
    public bool isEmployed { get; set; }
    public double LeaveAccruedDays { get; set; }
}
```



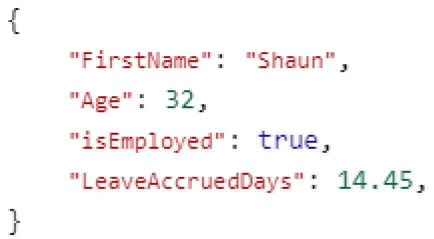
```
{
    "FirstName": "Shaun",
    "Age": 32,
    "isEmployed": true,
    "LeaveAccruedDays": 14.45,
}
```

JSON Creation

- There are various tools in most languages for creating JSON from objects in your code based upon Class objects.
- Many Web Frameworks and some database libraries will even handle this automatically for you.
- You can also write your own JSON manually as a string.
- In many languages though, if writing the JSON strings manually, you will need to escape out all the special characters.



```
public class Employee
{
    public string FirstName { get; set; }
    public int Age { get; set; }
    public bool isEmployed { get; set; }
    public double LeaveAccruedDays { get; set; }
}
```



```
{
    "FirstName": "Shaun",
    "Age": 32,
    "isEmployed": true,
    "LeaveAccruedDays": 14.45,
}
```

{\"FirstName\": \"Shaun\",\"Age\": 32,\"isEmployed\": true,\"LeaveAccruedDays\": 14.45}