Portfolio

Web Portfolio:

* Development diary.
* Web Domain and links.
* Show a representation of the work you have done.
* Current Knowledge.
* Strengths
* Weaknesses
* What you picked up and how you've improved.
* Dictionary of Technical Terms.

GCSE Examining board:

* AQA
* Edexcel
* OCR
* Pearson
* WJEC
* Nothern Ireland.

OCR GCSE Exam

* 40% Exam on Computer Systems.
* 40% Exam on Computational Thinking, Algorithms and Programming.
* 20% Programming projects.

GCSE Grades are now 9-1

Links

<http://www.ocr.org.uk/qualifications/gcse-computer-science-j276-from-2016/>

<http://qualifications.pearson.com/en/qualifications/edexcel-gcses/computer-science-2016.html>

<http://www.aqa.org.uk/subjects/ict-and-computer-science/gcse/computer-science-8520>

**Hardware.**

Comparing the brain and human body to the computer systems.

Inputs:

* Keyboard
* Mouse
* Sight
* Smell
* Touch
* Speech
* Hearing

Processing:

* Brain
* CPU

Output:

* Printers
* Bodily functions.

Storage:

* Hard drive- Magnetic Drives
* SSD Drives- Solid State Drives
* RAM- Random Access Memory
* ROM- Random Only Memory

History of Computers:

* BBC Computers had an operating system on a ROM.
* Commodore 64 and 128
* Sir Clive Sinclair ZX 81 Spectrum
* 5 1/4 inch floppy disk
* 3 1/2 inch floppy disk.
* Magnetic Tape Drives.

Hardware = Physical System

Software = The stuff to make it work.

Operating System = Windows 3.1, Windows and DOS in the old days.

Programming

Programming languages include:

* FORTRAN
* Assembly Language
* Machine Language
* DOS
* SQL
* Python 3.3.5 64 bit
* Pyscripter 2.5
* C++
* Object oriented Programming.
* C#

Places to programme:

* Code Academy
* Hour of code
* w3schools.com

Teaching Strategies

* Portable python.
* Portable pyscripter.
* Introduction to Python.

Links

<http://agbonline.co.uk/SKE.html>

Binary

Devices have two states 1 or 0.

|  |  |
| --- | --- |
| Denary | Binary |
| 0 | 0 |
| 1 | 001 |
| 2 | 010 |
| 3 | 011 |
| 4 | 100 |
| 5 | 101 |
| 6 | 110 |
| 7 | 111 |
| 8 | 1000 |
| 9 | 1001 |
| 10 | 1010 |
| 11 | 1011 |
| 12 | 1100 |

Binary is in Base 2.

For example 4 would be show as:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 |

**Teaching Ideas.**

Activity.

Give 5 people a card with the numbers 1,2,4,8,16 on. Say a set of numbers and get the students to work out which numbers are needed to make that number.

Computer Science Unplugged.

[JohnPhillipsJones.com](file:///C:\Catherine\0PGCEComputer\JohnPhillipsJones.com)

CD and Music- Reward good behaviour and working with choosing a winner each class that can bring in a CD to play next lesson.

Maths-aids.com

Little man computers

**Maths.**

101 = 10

102 = 10 x 10 = 100

103 = 10 x 10 x 10 = 1000

104 = 10 x 10 x 10 x 10 = 10000

Binary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 25 | 24 | 23 | 22 | 21 | 20 |
| 32 | 16 | 8 | 4 | 2 | 1 |

Base 8

|  |  |  |  |
| --- | --- | --- | --- |
| 83 | 82 | 81 | 80 |
| 512 | 64 | 8 | 1 |

Other.

* Denary
* Binary
* ASCII - American Standard Code for Characters.
* Hexadecimal

8 Bits = 1 Byte

4 Bits = 1 Nibble

Byte = 8 Bits

Kilobyte = 1024 Bits

Terabyte.

Petabyte.

Assembly Language consisted of:

* ADD
* SUB
* MUL
* DIV
* PRNT

Denary to Binary

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Denary | 1024 | 512 | 256 | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| 29 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 53 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| 74 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 555 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 752 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
|  |  |  |  |  |  |  |  |  |  |  |  |

Octal system Base 8.

|  |  |  |  |
| --- | --- | --- | --- |
| 83 | 82 | 81 | 80 |
| 512 | 64 | 8 | 1 |
|  |  |  |  |
| 4 | 0 | 7 | 3 |

Hexadecimal Base 16

|  |  |
| --- | --- |
| Denary | Hexadecimal |
| 0 | 0 |
| 1 | 1 |
| 2 | 2 |
| 3 | 3 |
| 4 | 4 |
| 5 | 5 |
| 6 | 6 |
| 7 | 7 |
| 8 | 8 |
| 9 | 9 |
| 10 | A |
| 11 | B |
| 12 | C |
| 13 | D |
| 14 | E |
| 15 | F |
|  |  |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 163 | 162 | 161 | 160 |
| 4096 | 256 | 16 | 1 |

Binary addition

0 + 0 = 0 carry 0

0 + 1 = 1 carry 0

1 + 0 = 1 carry 0

1 + 1 = 0 carry 1

1 + 1 + 1 = 1 carry 1

Binary Addition

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | 1 | 1 | 1 | 1 | 0 |
|  |  | 1 | 1 | 0 | 1 | 1 |
| Total | 1 | 1 | 1 | 0 | 0 | 1 |
| Carry | 1 | 1 | 1 | 1 | 0 |  |

Octal to denary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Octal Number | 512 | 64 | 8 | 1 | Denary Equivalent |
| 40738 | 4 | 0 | 7 | 3 | (4x512) + (0x64)+ (7x8) +(3 x 1) =2048+0+56+3 = 210710 |

Hexadecimal to Denary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Hexadecimal | 256 | 16 | 1 | Denary Equivalent |
| 61016 | 6 | 1 | 0 | (6x256)+ (1x 16) + (0x1) = 1536+16+0 = 155210 |

BIDMAS

Brackets

Indices

Division

Multiplication

Addition

Subtraction

Binary to Hexadecimal Conversion

Split Binary into sets of 4 numbers

|  |  |  |
| --- | --- | --- |
| 0111 | 0111 | 0111 |
| 7 | 7 | 7 |

Hexadecimal to Binary Conversion

|  |  |  |
| --- | --- | --- |
| F (15) | 3 | C(12) |
| 1111 | 0011 | 1100 |

Binary to Octal Conversion

Split into sets of 3 numbers

|  |  |  |  |
| --- | --- | --- | --- |
| 001 | 011 | 011 | 101 |
| 1 | 3 | 3 | 5 |

Octal to Binary Conversion

|  |  |  |  |
| --- | --- | --- | --- |
| 5 | 6 | 0 | 4 |
| 101 | 110 | 000 | 100 |

Minecraft and Raspberry Pi

Items needed:

* Monitor
* Pi version 3
* USB Keyboard
* USB Mouse
* HDMI Cable
* Network Cable
* Wifi Dongle
* Pi Power supply cable

Come with pre loaded software NOOBS

* Operating system
* Minecraft
* Python

**Places to buy.**

<http://cpc.farnell.com/>

<https://thepihut.com/>

richwalk.home@gmail.com

**Books**

Adventures in Minecraft.

Building a Computer.

Building a Computer Presentation.ppt



Parts of Computer.

* Hard Drive,
* CPU
* Processor
* Motherboard with CMOS Batterr
* Graphics Card like NVIDIA.
* RAM 512 always match the pairs.
* Sound Card
* CD/ DVD Drive.
* Modems in the old days
* Network connection.
* Power supply

Computer parts can be brought from:

* DABS
* Ebuyer
* PCPartpicker.co.uk

**Teaching Ideas**

"Cost up a computer with all the component parts."

Watch Imitation game (Alan Turing) with Benedict Cumberbatch.

Computer builders of history.

* Charles Babbage and the analytical engine.
* John von neumann.
* Ada Lovelace.
* Alan Turing and Bletchley Park.

Javascript.





Flowol

Lesson 1 aimed at Year 7.

What is a control system?

"What we are going to do is build a control system."

Fundamentals.

A control system is a system to control something.

Input - Process- Output

Inputs

* Make a circuit.
* Passive Infer Red
* Sirens
* Fire doors closing.

"Can you tell me another control system?"

* Fridge
* Sensors
* Rides at a Theme Park
* Train, Crossroads
* Sensors on Train.

Improving safety by removing people from the equation.

Definition is Input - Control - System- Output.

Lesson 2.

Pelican Crossing example.

PE Pedestrian

LI Lighting

CON Control

There are 27 things that happen in a pelican crossing.

* On green
* If no visits stay on green.
* Press to cross.
* Press button then wait.
* Little red man
* Lights to amber
* Beep beep when lights on red.

Using Flowal.

File

* new
* open
* save

Settings

* Options

Edit

* Undo
* Copy
* Paste
* Redo.

**Teaching Strategies.**

1. Show a little bit at a time. Then get the students to do it.

2. My turn, Your turn.

3. Wait for silence, whilst saying "We'll just wait for you to be quiet".

4. Quiet please ..... This is the sound I want to hear...

**Flowol builds a computer control system using Mimic Software**

Zebra Crossing example with Bolisha beacons.



Lighthouse



Level Crossing.

To add a new Mimic.

* Settings
* Mimic
* Add .
* Search for folder

Python.

Download Python 3.3.5

* Download Pyscriptor 2.5

Lesson 1.

Printing to the page

* print(3, "This is the number 3")
* print(3, " This is the number 3", 15+9)

Maths

* - subtract
* + addition
* multiplication
* / division
* print(3\*412)

Printing things 5 times

* print ("Andy" \*5)
* print ("Andy\n" \*5)

Name variable

* name= input("What's your name?")
* print(name,"is a lovely name")

Exercise Hints

Exercise 1.

Real numbers 3.8

String number 3.8

Booleon numbers are True and False

Perform Arithmetics.

Exercise 2.

% is for formatting.

%.2f% = 2 decimal places

%.3f% = 3 decimal places

Exercise 3.

While loop to repeat things

== is equal to .

**Teaching Ideas.**

* Agbonline Teaching Exercises.
* Using python.com
* 20 code Challenges( complete for GCSE 9-1 )
* Python Exercises.
* Introduction to Python.

**Banner program.**

#ask the user for their name

name=input("Please enter your name")

#dashes

print("-" \*(len(name)+6))

print(".: " +name+ " :.")

#print the second row of dashes

print("-" \*(len(name)+6))

Start

Get Name

Prepare Banner

Output Banner

End

**Average Calculator.**

#Average number calculator

#declare array

num\_array = list()

# enter number of numbers to average

num = input("Enter how many numbers you would like to find the average for:")

# enter numbers into array

for i in range(int(num)):

n = input("Enter numbers into array:")

num\_array.append(int(n))

print ("ARRAY: ",num\_array)

#check how many numbers where entered.

print("Number of numbers entered",num)

#add up the numbers in the array.

average = sum(num\_array)/len(num)

#declare num as a float.

number=float(num)

#divide average by number.

mean = average/number

#Show total of array.

print ("Total of Array:",average)

#show average of array.

print ("Average of numbers:",mean)

print("Thank you and goodnight")

**Image Representation.**

History of Images

TV used Electron Beams.

It used R G B colours and coordinates of X across and Y down.

Computer Thinking is in Binary and Hexadecimal.

0

1

2

3

4

5

6

7

8

9

10 - A

11 - B

12 - C

13 - D

14 - E

15 - F

**Pixel Conversion Exercise.**

Use a picture, turn it into excel sheet, change the picture just using the numbers.

C:\Catherine\00UCASITT\pixel\Binary and Bitmapped Images-revision-8\Windows-4June2012-Pixel Spreadsheet\release

1. Find a picture.

2. Open the TestingCVandGUI application.

3. Upload a photo.

4. Convert to Comma delimited file.

5. Open in Excel

6. Change the numbers around.

7. Save

8. Re- upload to Application

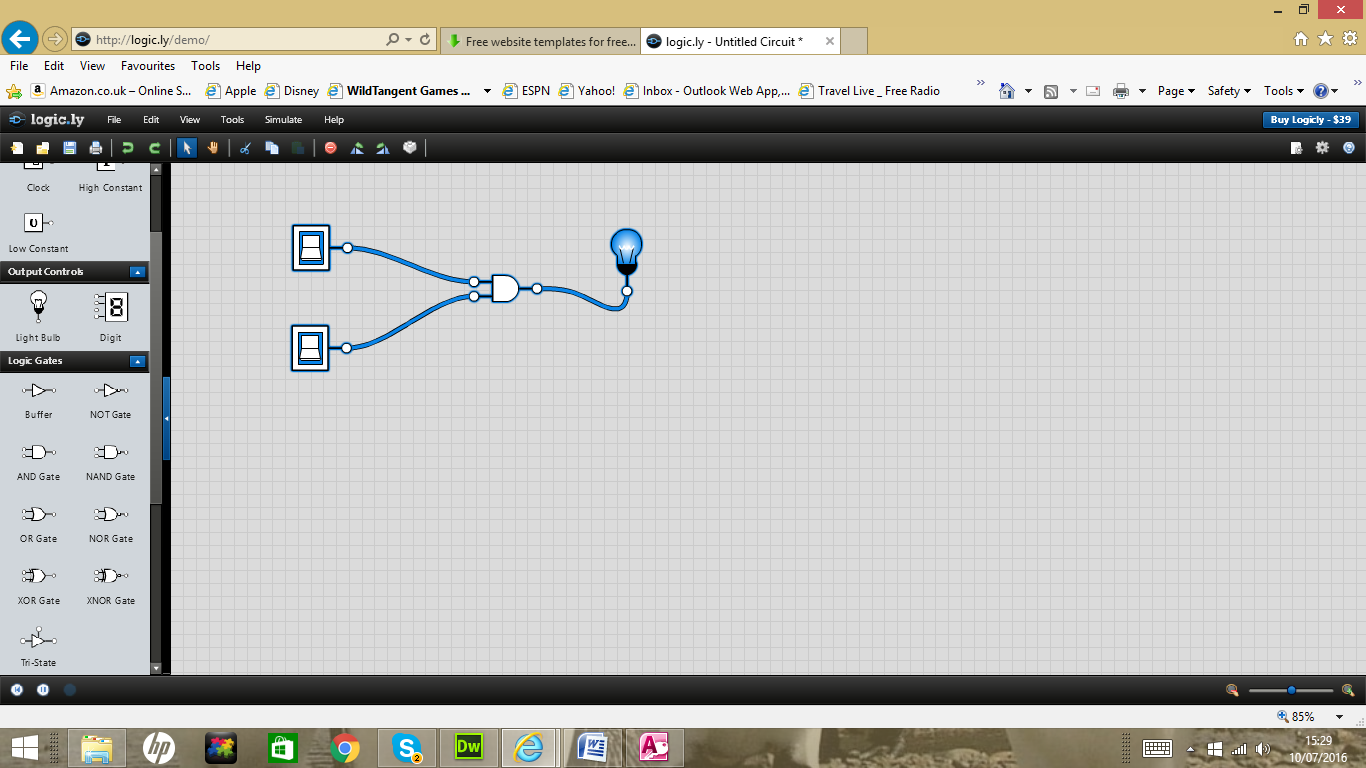
9. See the changes to the application.

10. Repeat when changing colours.

Logic Gates.

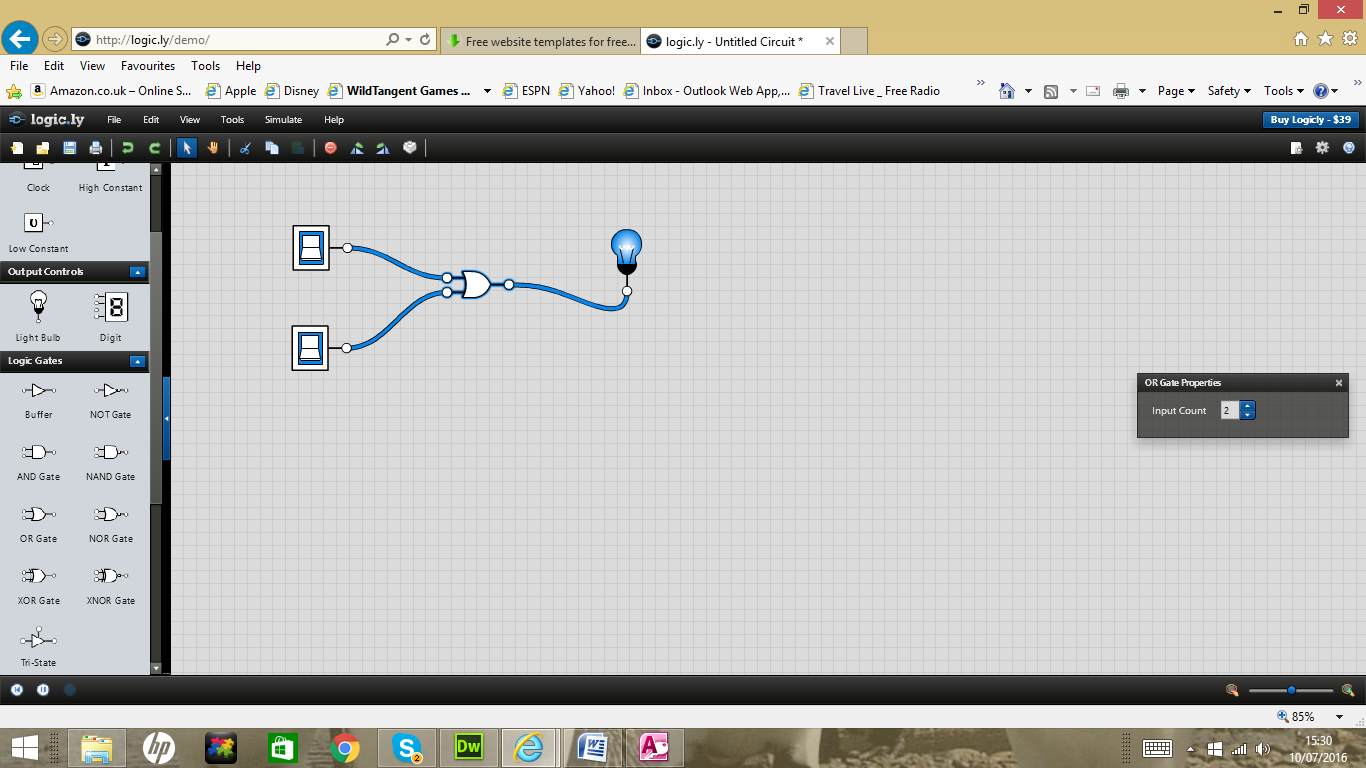
http://logic.ly/demo/

AND Gate.



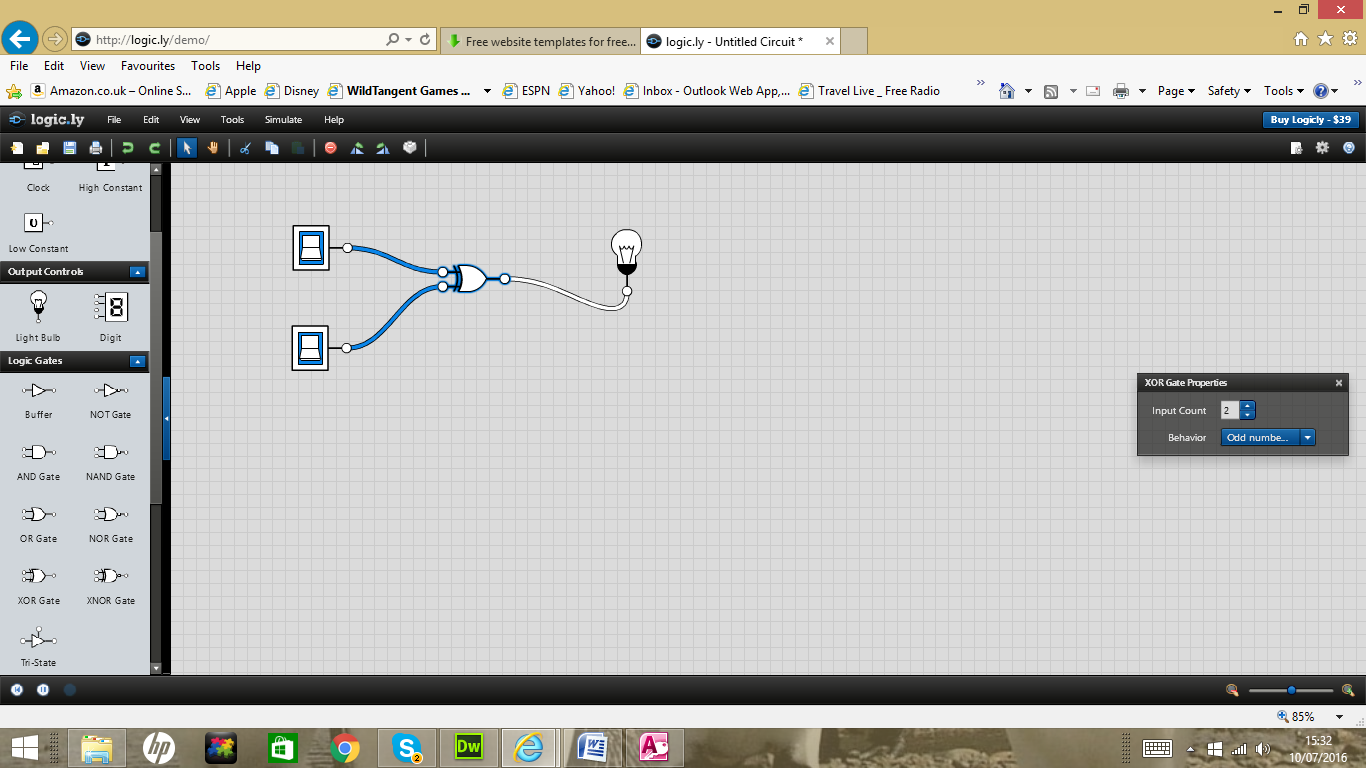
|  |  |  |
| --- | --- | --- |
| A | B | X = A AND B |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

OR Gate



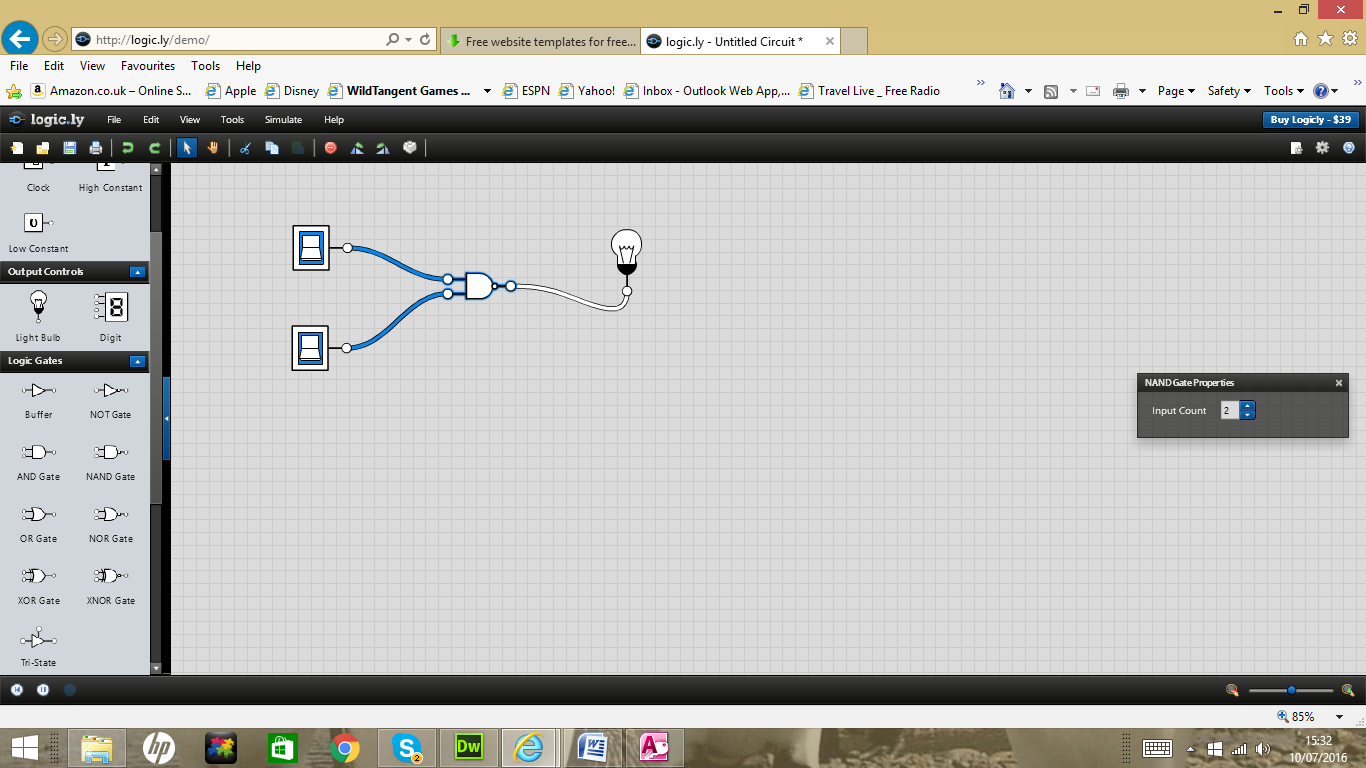
|  |  |  |
| --- | --- | --- |
| A | B | X= A OR B |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

XCLUSIVE OR Gate



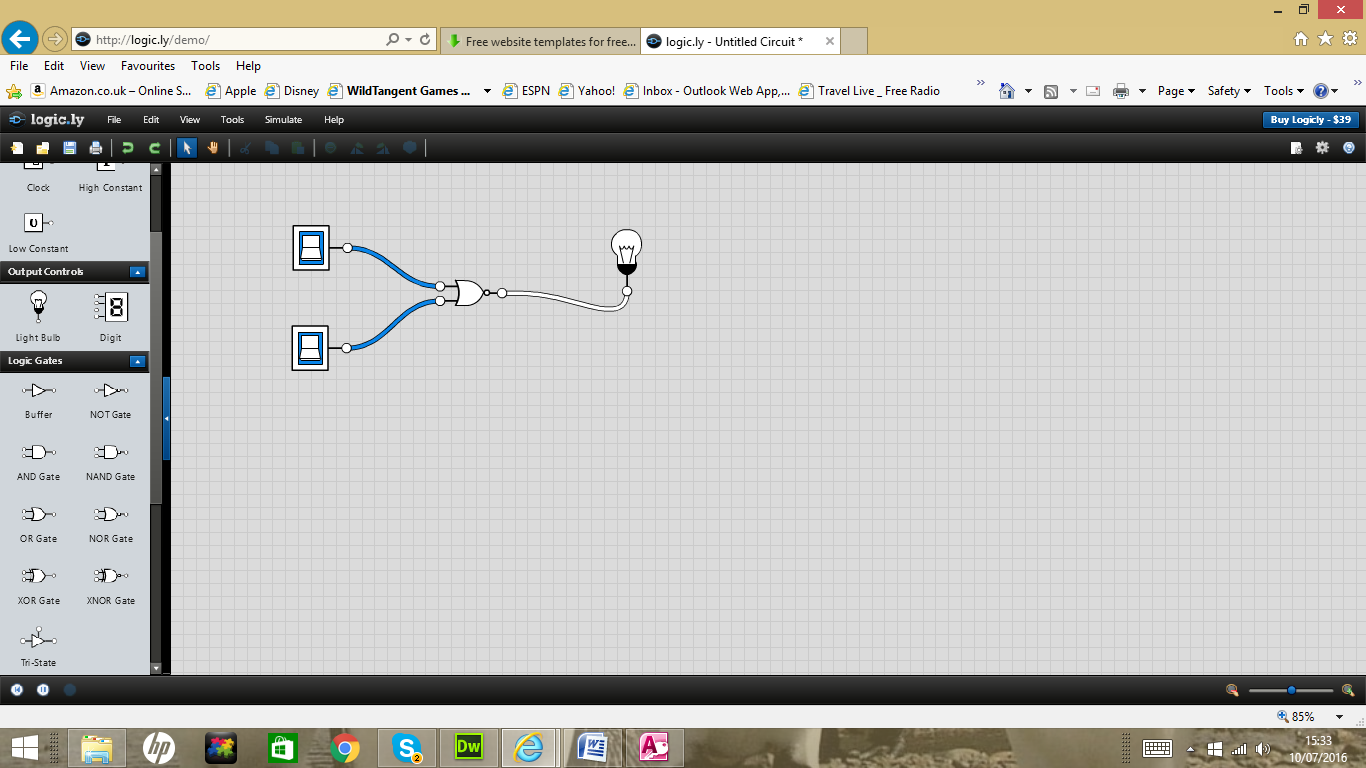
|  |  |  |
| --- | --- | --- |
| A | B | X |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

NAND Gate



|  |  |  |  |
| --- | --- | --- | --- |
| A | B | C | X |
| 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

NOR Gate



|  |  |  |
| --- | --- | --- |
| A | B | X |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

**Drawing Flowcharts for Python**

Google Drawing and Word Drawing, Gliffy and Creatly can be used to create generic flowcharts.

Flowcharts Basic Symbols.

Start

Decision

Data

Process

Document

Pre-defined Process

We looked at the Average Calculator in Python.

First we created an average calculator flowchart.

Get Number

Yes

Is it 999

No

Calculate and output Average

Increment Count

Output Average

Add to total

Stop

Average Calculator Python (Array and non array versions)

i=0

c=0

t=0

#while i !="999":

# i = input("Enter numbers or '999' to finish:")

# if i!="999":

# c+=1

# t=t+float(i)

#ave= t/c

#print("Total = ",i)

#print ("Average%5.2f"%ave)

#Average number calulator

#declare array

num\_array = list()

# enter number of numbers to average

while i !="999":

i = input("Enter numbers into array:")

if i!="999":

c+=1

num\_array.append(int(i))

print ("ARRAY: ",num\_array)

#add up the numbers in the array.

average = sum(num\_array)

#declare num as a float.

#number=float(num)

#divide average by number.

mean = average/c

#Show total of array.

print ("Array:",num\_array)

#show average of array.

print ("Average of numbers:",mean)

print("Thank you and goodnight")

Roll a dice.

#life decisions

import random

decisions=["Stay in Bed","Go for a walk","Go clothes shopping","Hang out with friends", "Do some programming"]

print("What to do today:", random.choice(decisions))

#roll a dice

import random

print("Roll a six to win")

number=random.randrange(1,7)

print("Roll is",number)

**Guess the roll of the dice**

#\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

def ask\_user(prompt):

ug=0

while not(1<=ug<=6):

try:

ug=int(input(prompt))

if not(1<=ug<=6):

raise ValueError()

except ValueError:

prompt="Invalid entry. It must be 1 to 6."

return ug

#\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

spin\_count=0

wins=0

while True:

users\_guess = ask\_user("Guess the roll of the dice")

number=random.randrange(1,7)

spin\_count=spin\_count+1

print("Number of spins:{}. The roll is:{}".format(spin\_count, number))

if number==users\_guess:

wins+=1

win\_percent=wins/spin\_count

print("This is a win number{}!. You win percentage is {:.2f}".format(wins,win\_percent))

else:

print("You lose")

**Reading a file**

#Read my Timetable

#-----------------

def readfile(day):

filename=day+".txt"

channel=open(filename,"r+")

lesson=channel.readlines()

channel.close()

return(lesson)

#-------------------

day=input("Day of Week")

lesson=readfile(day)

print(lesson)

print("Day- ",day,"\n")

for i in lesson:

print(i)

**Python Controlled Assessment Practice**

score = 0

qcount= 0

import random

import operator

decisions=[10,20,30,40,50,2,3,4,5,6,7,8,9,15,25,35,45]

operators=["+","-","\*"]

#ops={"+": operator.add, "-": operator.sub, "\*":operator.mul}

name = (input("Please enter your name :"))

while qcount != 10:

qcount=qcount+1

number1=random.choice(decisions)

number2=random.choice(decisions)

ops=random.choice(operators)

question=str(number1)+ops+str(number2)

realanswer=eval(question)

#print(realanswer)

studentanswer=int(input("Question number "+ str(qcount) + "\n" + question +"= ?"))

#print(studentanswer)

if realanswer==studentanswer:

score=score+1

print("Your correct")

else:

print("Sorry your answer is incorrect")

print("++++++++++++++++++++++++++++++")

print("Name:",name)

print("Score:",score, " out of 10")

print("+++++++++++++++++++++++++++++++")

def readfile(class):

filename=class+".txt"

channel=open(filename,"r+")

results=channel.readlines()

channel.close()

return(class)

#-------------------

def writefile(class):

line1 = raw\_input("line 1: ")

line2 = raw\_input("line 2: ")

line3 = raw\_input("line 3: ")

print "I'm going to write these to the file."

filename.write(line1)

filename.write("\n")

filename.write(line2)

filename.write("\n")

filename.write(line3)

filename.write("\n")

def closefile(class):

Print "And finally, we close it."

target.close()

class=input("Please enter a class")

results=readfile(class)

print(results)

Pick Random number

Teaching Ideas for Flowcharts

www.coolmath-games.com

Water Jars

Tower of Hanoi

Python Solutions.

Python exercises 1.

Python exercises 2.

Python exercises 3.

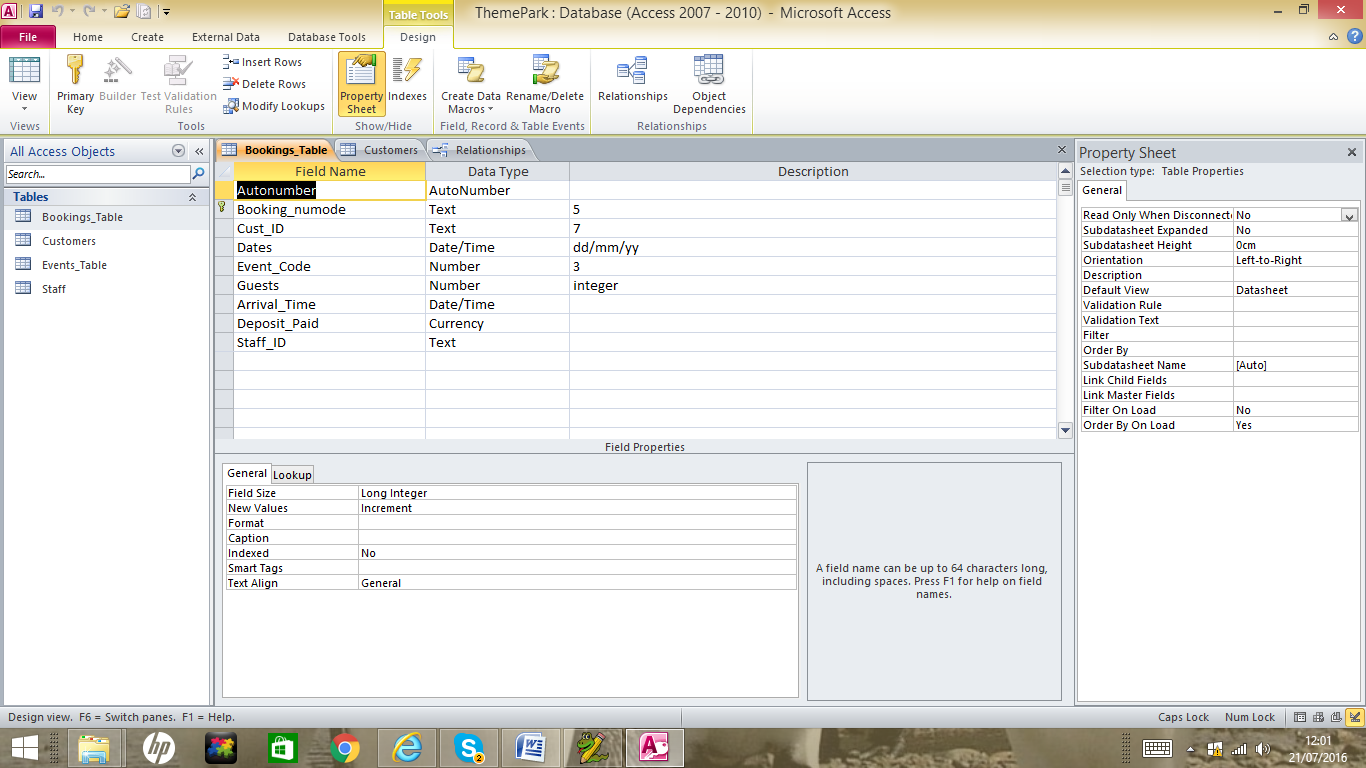
Lynda.com enter Staffs.ac.uk enter login details sm950723

**Databases and SQL.**

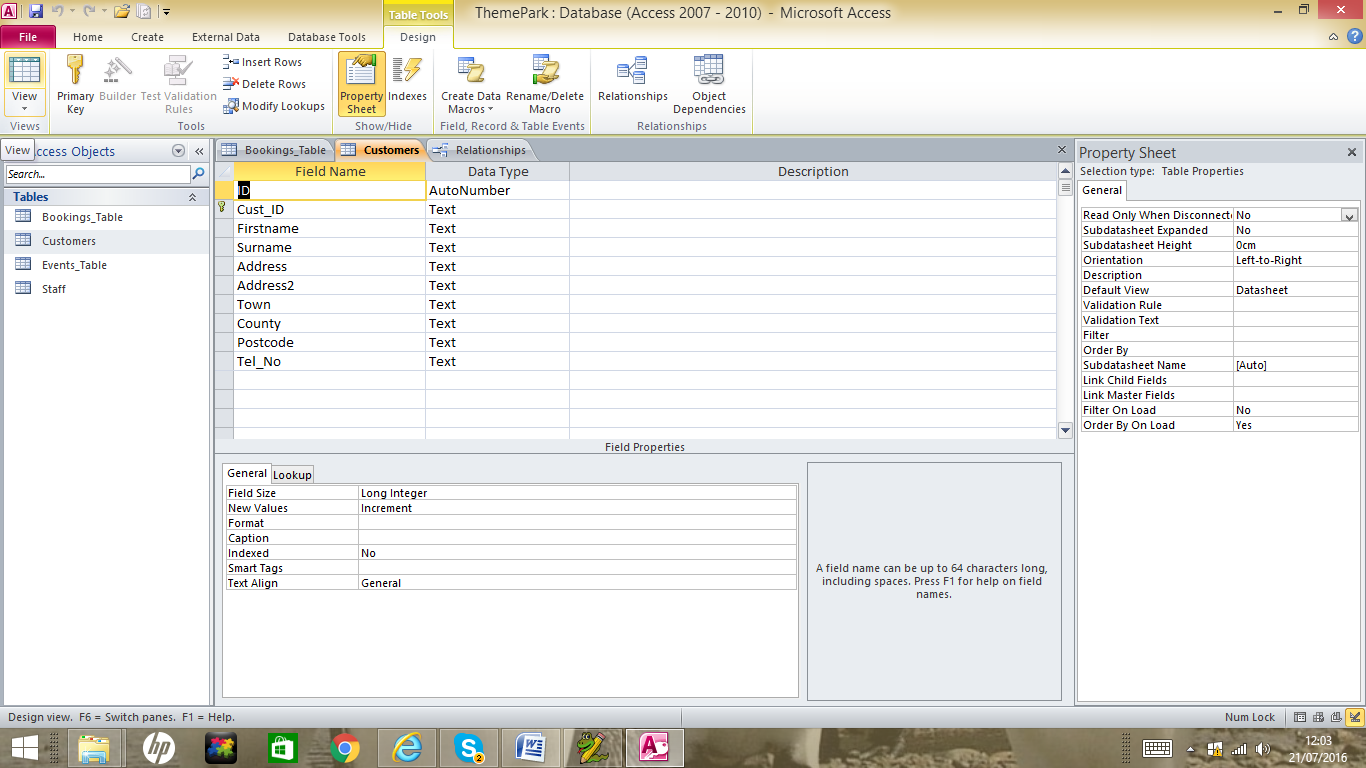
Access Database

Events Database

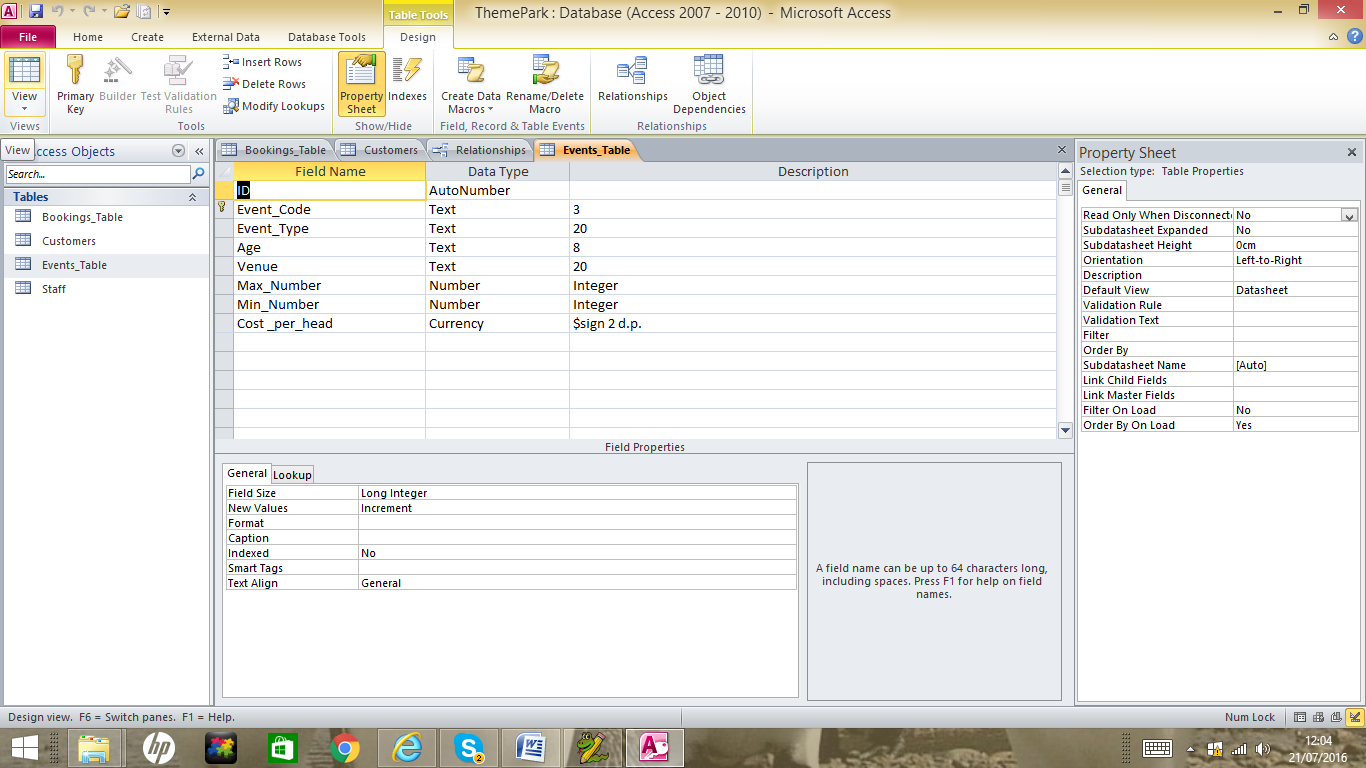
Create bookings table



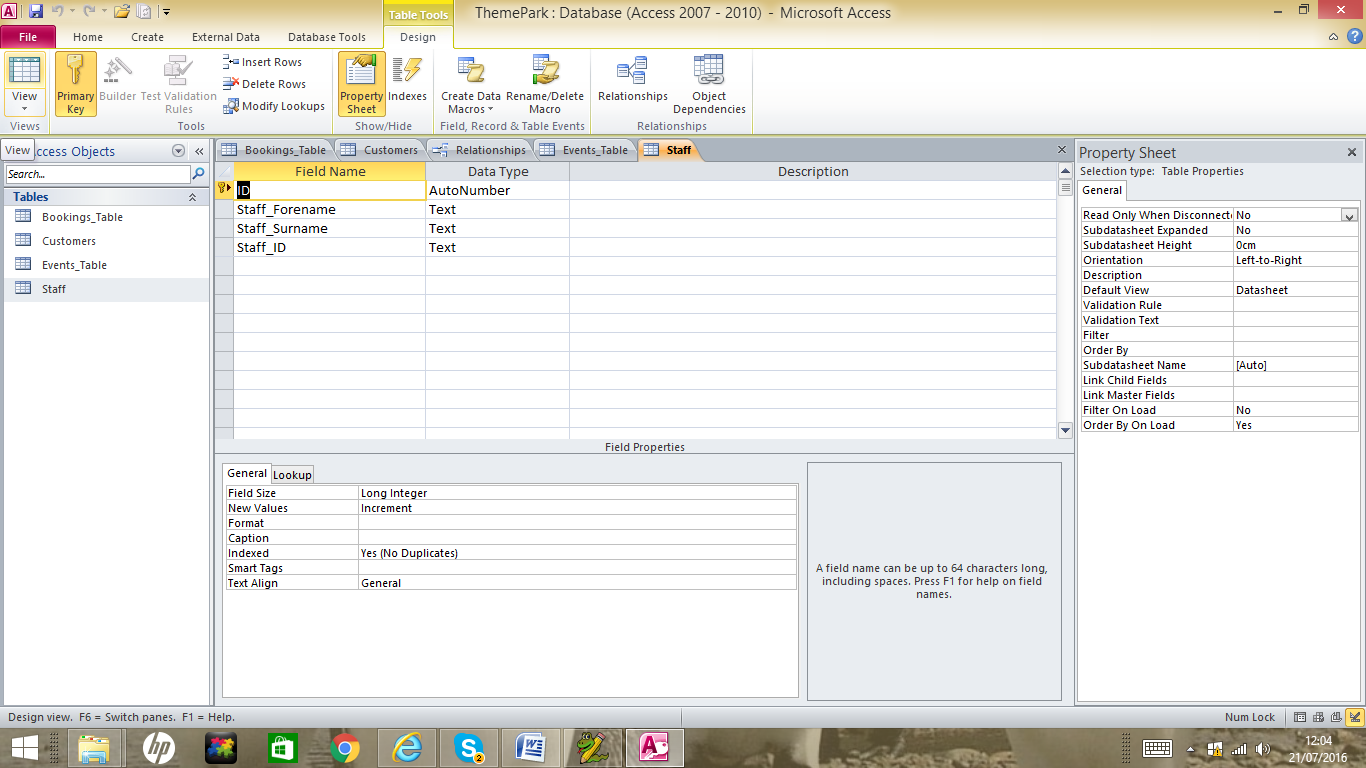
Create Customers table



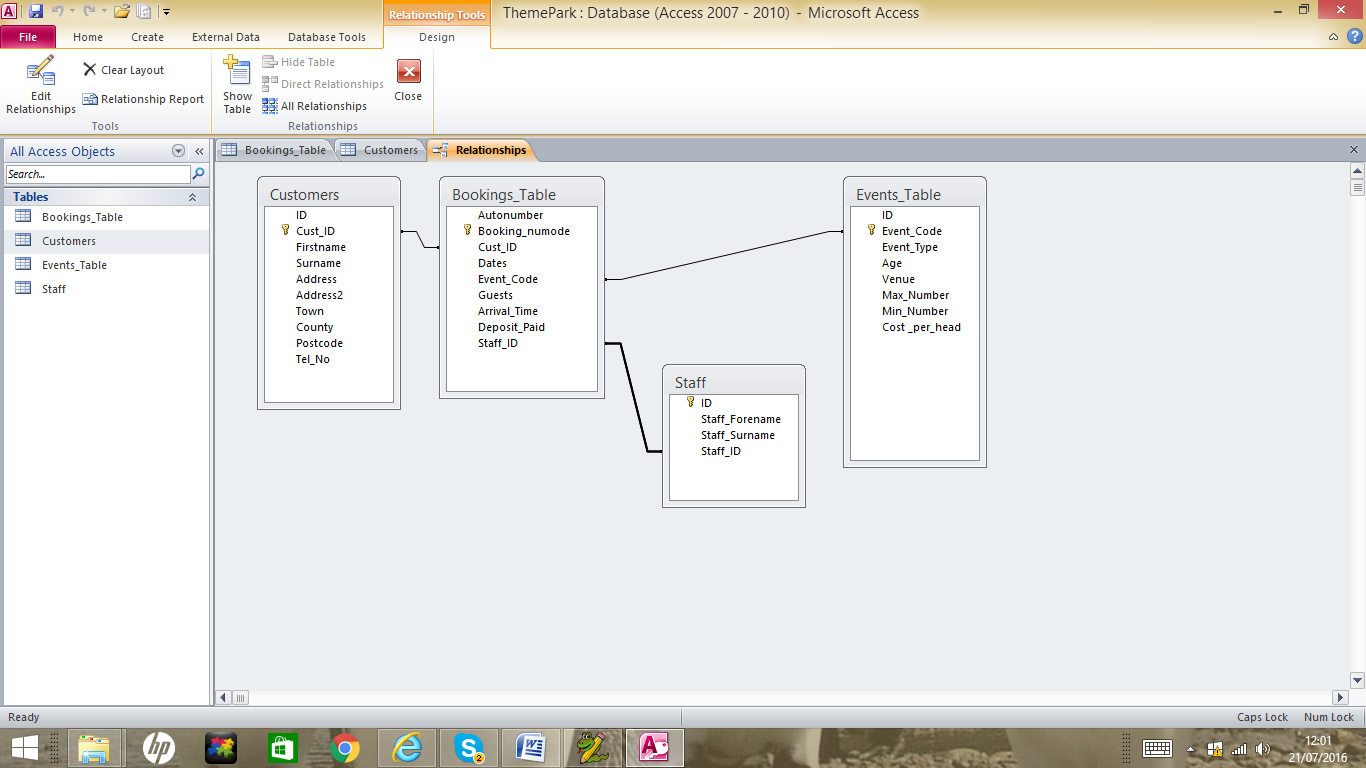
Create Events table



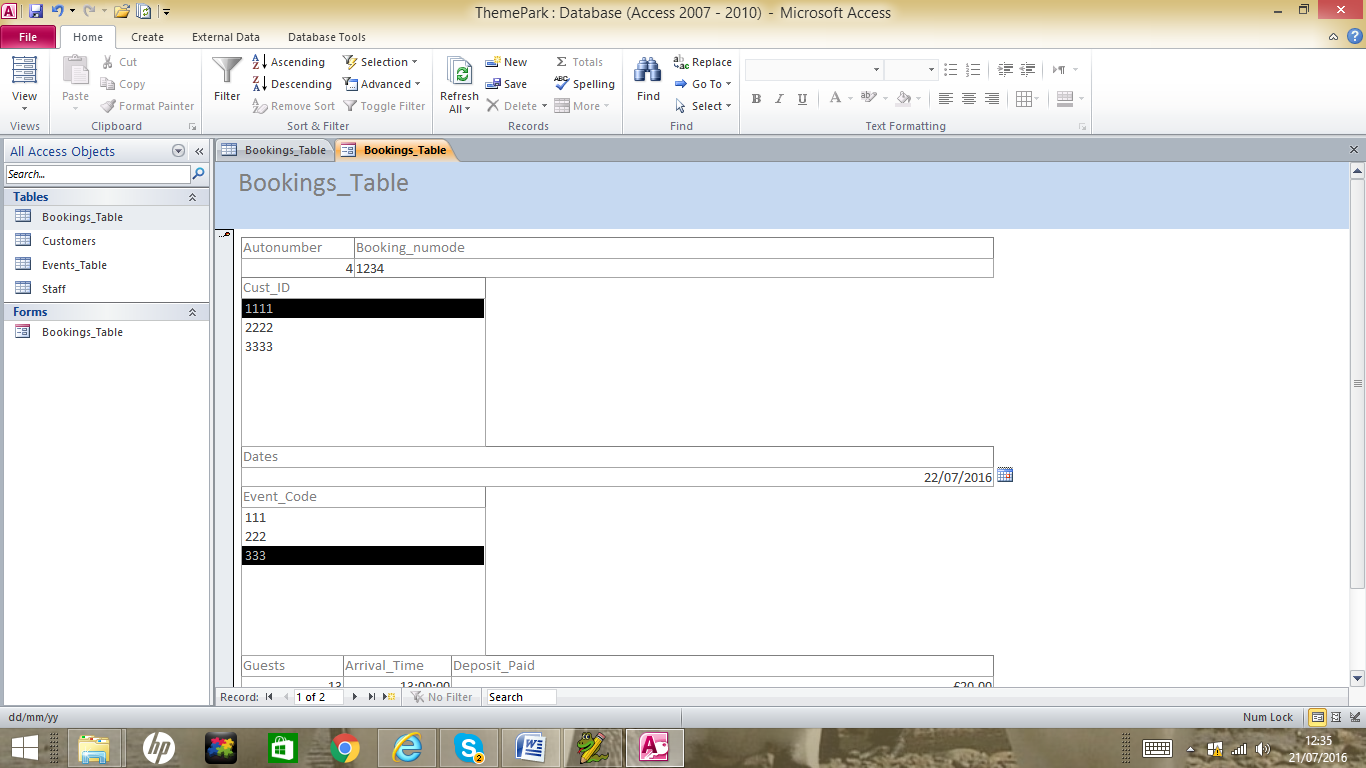
Create Staff table



Create a relationship between the tables

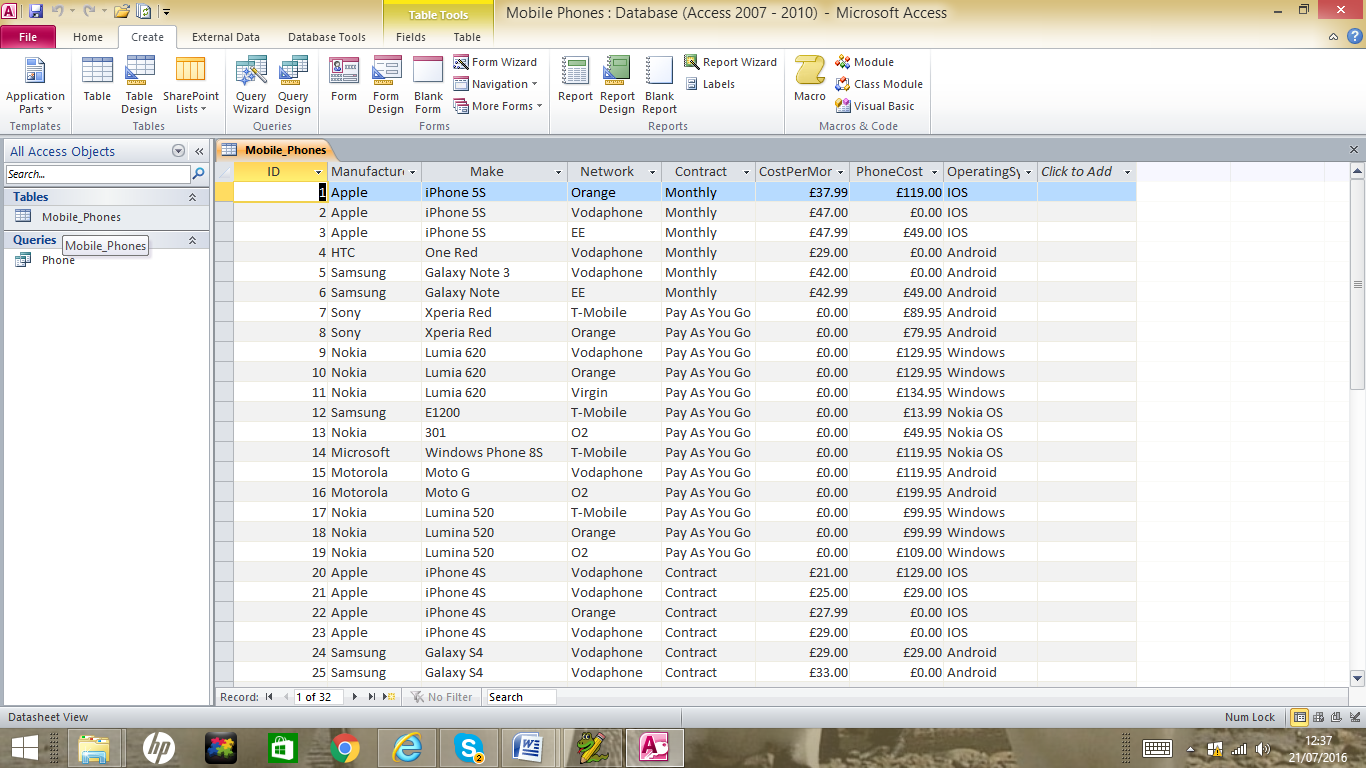


Booking Form

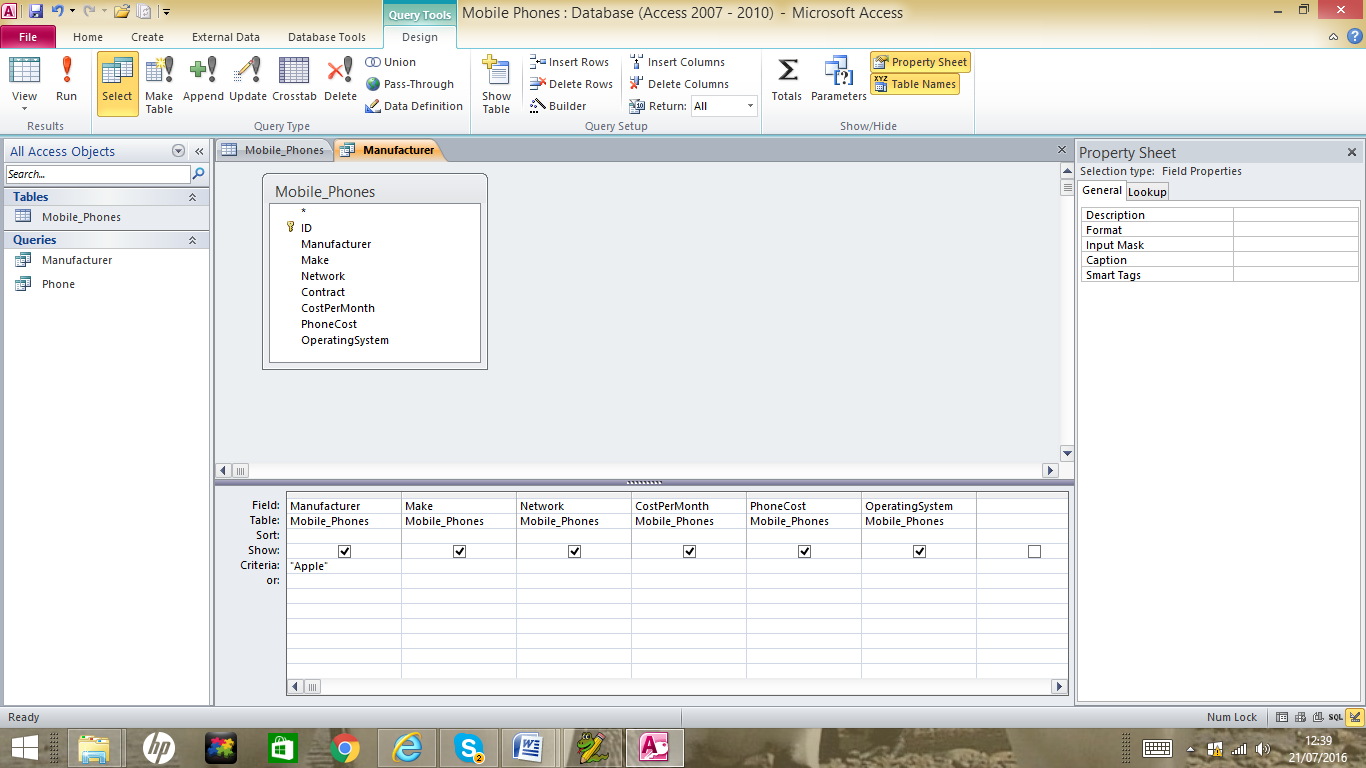


**Running queries with the Mobile\_Phone Database**

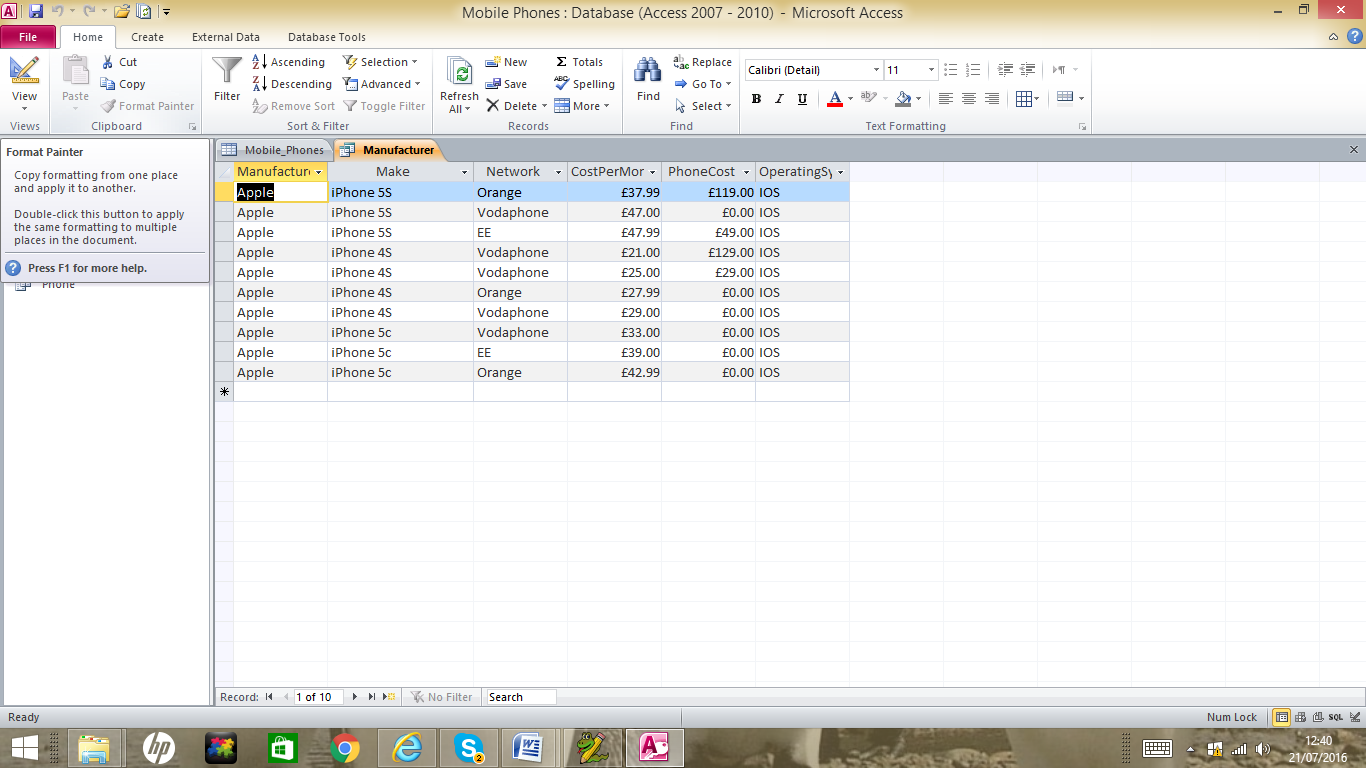
Mobile Phone Table



Manufacturer Query



Run it



right click on the query and go to SQL View.

SELECT Mobile\_Phones.Manufacturer, Mobile\_Phones.Make, Mobile\_Phones.Network, Mobile\_Phones.CostPerMonth, Mobile\_Phones.PhoneCost, Mobile\_Phones.OperatingSystem

FROM Mobile\_Phones

WHERE (((Mobile\_Phones.Manufacturer)="Apple"));

SQL Statements

|  |  |
| --- | --- |
| **SQL Statement** | **Syntax** |
| AND / OR | SELECT column\_name(s) FROM table\_name WHERE condition AND|OR condition |
| ALTER TABLE | ALTER TABLE table\_name  ADD column\_name datatype  or  ALTER TABLE table\_name  DROP COLUMN column\_name |
| AS (alias) | SELECT column\_name AS column\_alias FROM table\_name  or  SELECT column\_name FROM table\_name  AS table\_alias |
| BETWEEN | SELECT column\_name(s) FROM table\_name WHERE column\_name BETWEEN value1 AND value2 |
| CREATE DATABASE | CREATE DATABASE database\_name |
| CREATE TABLE | CREATE TABLE table\_name ( column\_name1 data\_type, column\_name2 data\_type, column\_name2 data\_type, ... ) |
| CREATE INDEX | CREATE INDEX index\_name ON table\_name (column\_name)  or  CREATE UNIQUE INDEX index\_name ON table\_name (column\_name) |
| CREATE VIEW | CREATE VIEW view\_name AS SELECT column\_name(s) FROM table\_name WHERE condition |
| DELETE | DELETE FROM table\_name WHERE some\_column=some\_value  or  DELETE FROM table\_name  (**Note:**Deletes the entire table!!)  DELETE \* FROM table\_name  (**Note:**Deletes the entire table!!) |
| DROP DATABASE | DROP DATABASE database\_name |
| DROP INDEX | DROP INDEX table\_name.index\_name (SQL Server) DROP INDEX index\_name ON table\_name (MS Access) DROP INDEX index\_name (DB2/Oracle) ALTER TABLE table\_name DROP INDEX index\_name (MySQL) |
| DROP TABLE | DROP TABLE table\_name |
| EXISTS | IF EXISTS (SELECT \* FROM table\_name WHERE id = ?) BEGIN --do what needs to be done if exists END ELSE BEGIN --do what needs to be done if not END |
| GROUP BY | SELECT column\_name, aggregate\_function(column\_name) FROM table\_name WHERE column\_name operator value GROUP BY column\_name |
| HAVING | SELECT column\_name, aggregate\_function(column\_name) FROM table\_name WHERE column\_name operator value GROUP BY column\_name HAVING aggregate\_function(column\_name) operator value |
| IN | SELECT column\_name(s) FROM table\_name WHERE column\_name IN (value1,value2,..) |
| INSERT INTO | INSERT INTO table\_name VALUES (value1, value2, value3,....)  *or*  INSERT INTO table\_name (column1, column2, column3,...) VALUES (value1, value2, value3,....) |
| INNER JOIN | SELECT column\_name(s) FROM table\_name1 INNER JOIN table\_name2  ON table\_name1.column\_name=table\_name2.column\_name |
| LEFT JOIN | SELECT column\_name(s) FROM table\_name1 LEFT JOIN table\_name2  ON table\_name1.column\_name=table\_name2.column\_name |
| RIGHT JOIN | SELECT column\_name(s) FROM table\_name1 RIGHT JOIN table\_name2  ON table\_name1.column\_name=table\_name2.column\_name |
| FULL JOIN | SELECT column\_name(s) FROM table\_name1 FULL JOIN table\_name2  ON table\_name1.column\_name=table\_name2.column\_name |
| LIKE | SELECT column\_name(s) FROM table\_name WHERE column\_name LIKE pattern |
| ORDER BY | SELECT column\_name(s) FROM table\_name ORDER BY column\_name [ASC|DESC] |
| SELECT | SELECT column\_name(s) FROM table\_name |
| SELECT \* | SELECT \* FROM table\_name |
| SELECT DISTINCT | SELECT DISTINCT column\_name(s) FROM table\_name |
| SELECT INTO | SELECT \* INTO new\_table\_name [IN externaldatabase] FROM old\_table\_name  *or*  SELECT column\_name(s) INTO new\_table\_name [IN externaldatabase] FROM old\_table\_name |
| SELECT TOP | SELECT TOP number|percent column\_name(s) FROM table\_name |
| TRUNCATE TABLE | TRUNCATE TABLE table\_name |
| UNION | SELECT column\_name(s) FROM table\_name1 UNION SELECT column\_name(s) FROM table\_name2 |
| UNION ALL | SELECT column\_name(s) FROM table\_name1 UNION ALL SELECT column\_name(s) FROM table\_name2 |
| UPDATE | UPDATE table\_name SET column1=value, column2=value,... WHERE some\_column=some\_value |
| WHERE | SELECT column\_name(s) FROM table\_name WHERE column\_name operator value |

**Teaching Ideas for Databases and SQL.**

www.w3schools.co.uk

**Psuedo Code.**

Psuedo Code is another way of designing a programme. It allows you to work out the structure and even looping structure of a programme using normal English words.

**Commoncraft**

Use Commoncraft to create a video on "What is Secondary Storage?"

Either using drawings or a powepoint.

Commoncraft example document.

https://www.commoncraft.com/

**What is secondary storage?**

https://www.youtube.com/watch?v=wu3tCNlRUMk

We looked at what is secondary storage.

**Architecture, Machine Code and Assembly Language**

Machine code can be used on two websites these are:

http://robowriter.info/little-man-computer/bo

http://peterhigginson.co.uk/LMC/

Some LMC instructions on agbonline.co.uk are:

|  |  |  |
| --- | --- | --- |
| Instruction | Mnemonic | MachineCode |
| Load | LDA | 5xx |
| Store | STA | 3xx |
| Add | ADD | 1xx |
| Subtract | SUB | 2xx |
| Input | INP | 901 |
| Output | OUT | 902 |
| End | HLT | 000 |
| Branch if zero | BRZ | 7xx |
| Branch if zero or positive | BRP | 8xx |
| Branch always | BRA | 6xx |
| Data storage | DAT |  |

agbonline.co.uk

Basic Hello World Example

|  |  |
| --- | --- |
| 901 | In |
| 902 | Output |

**A Basic 2 number addition calculator.**

|  |  |  |  |
| --- | --- | --- | --- |
| Machine Code | Instructions | Line Number | Instruction |
| 901 | In | 00 | INP |
| 399 | Sto 99 | 01 | STA 99 |
| 901 | In | 02 | INP |
| 199 | In | 03 | ADD 99 |
| 902 | Out | 04 | OUT |
| 000 | Cob | 05 | HLT |
|  |  |  |  |

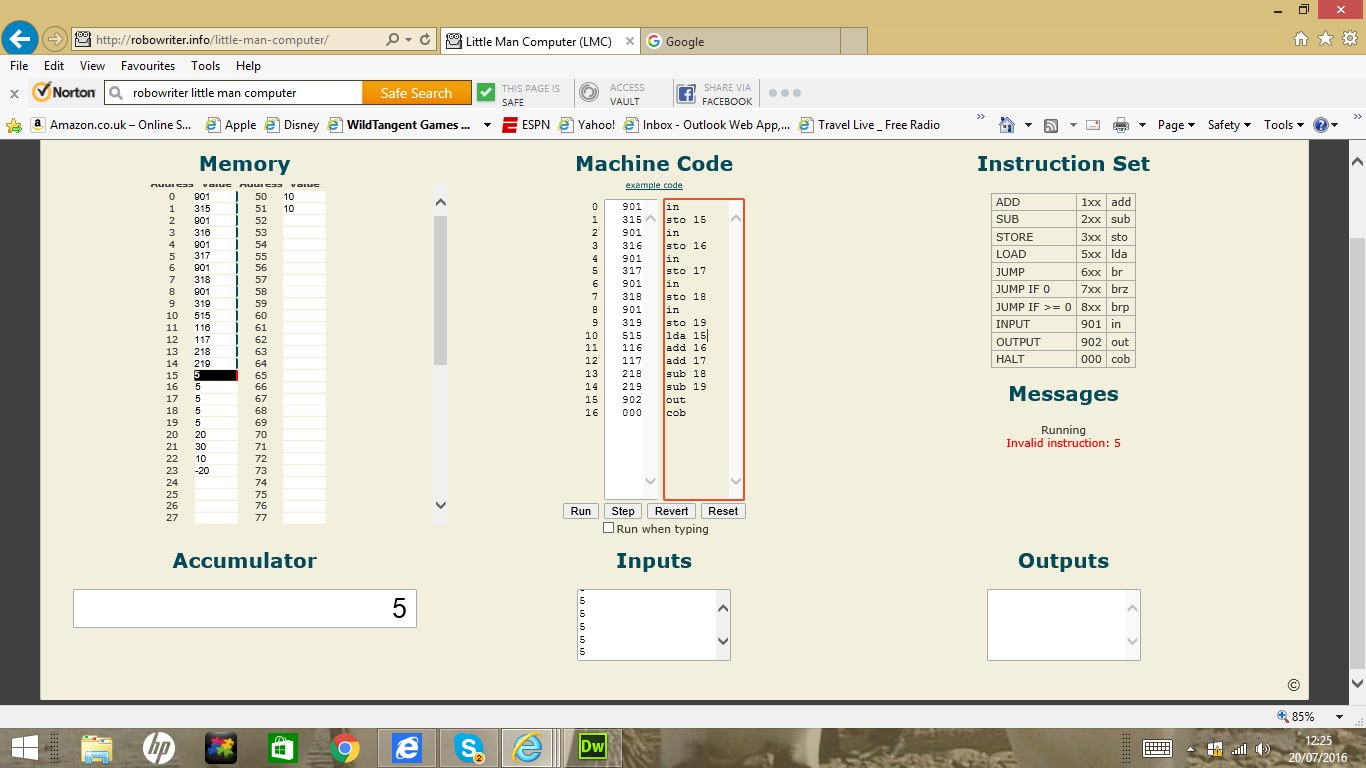
**Input 3 numbers.**

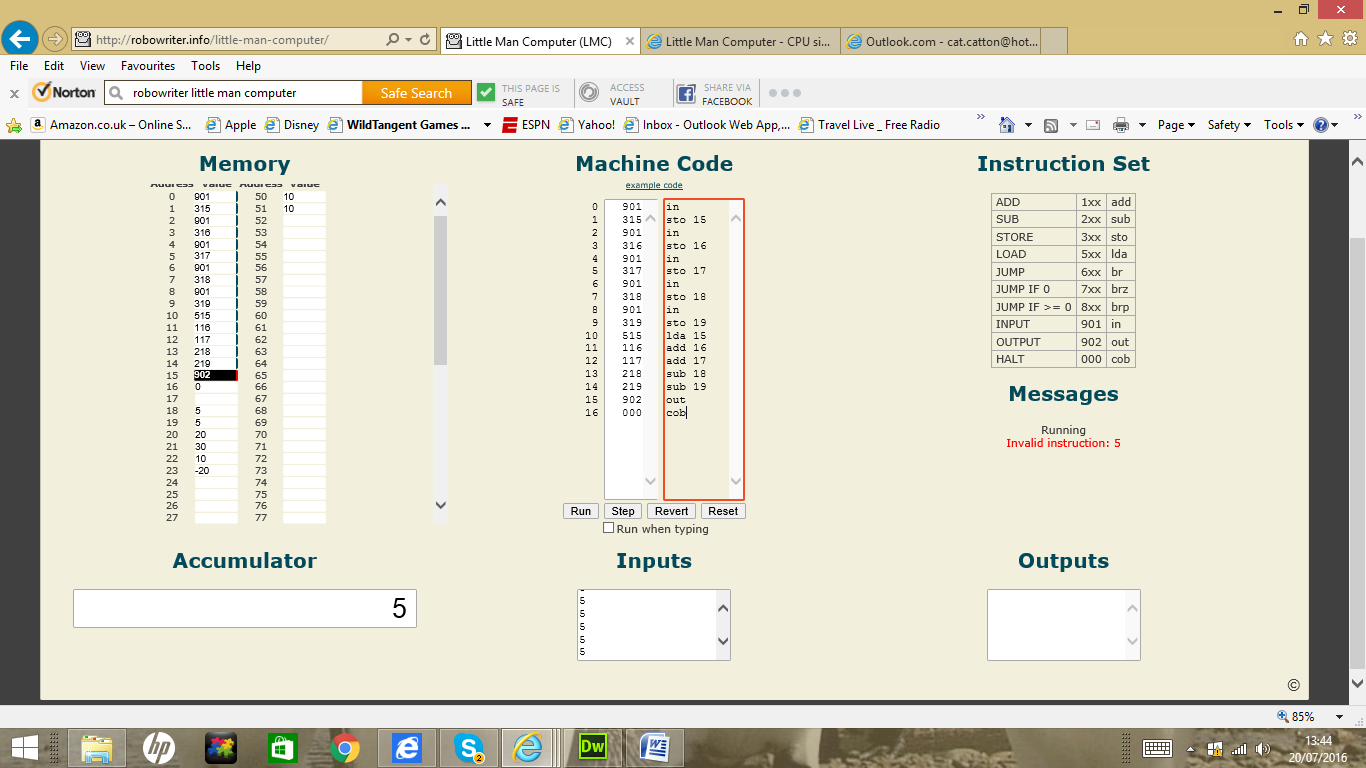
|  |  |
| --- | --- |
| 901 | In |
| 315 | Sto 15 |
| 901 | In |
| 115 | Add 15 |
| 316 | Sto 15 |
| 901 | In |
| 116 | Add 16 |
| 318 | Sto 18 |
| 902 | Out |
| 000 | Cob |
|  |  |

**Input 3 numbers and subtract 2 numbers.**

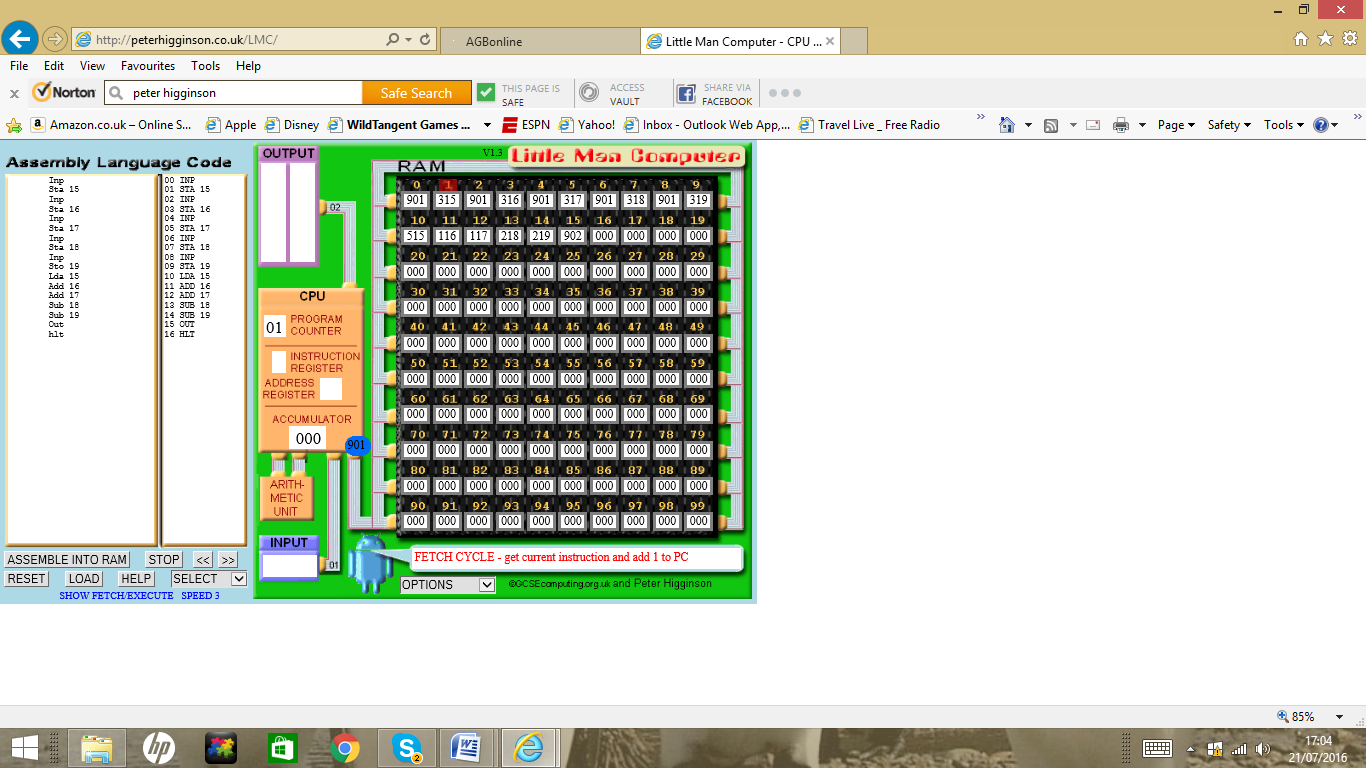
|  |  |  |
| --- | --- | --- |
| Location of instruction. | Instruction in location. | Instruction. |
| 0 | 901 | In |
| 1 | 315 | Sto 15 |
| 2 | 901 | In |
| 3 | 316 | Sto 16 |
| 4 | 901 | In |
| 5 | 317 | Sto 17 |
| 6 | 901 | In |
| 7 | 318 | Sto 18 |
| 8 | 901 | In |
| 9 | 319 | Sto 19 |
| 10 | 515 | Lda 15 |
| 11 | 116 | Add 16 |
| 12 | 117 | Add 17 |
| 13 | 218 | Sub 18 |
| 14 | 219 | Sub 19 |
| 15 | 902 | Out |
| 16 | 000 | Cob |

http://robowriter.info/little-man-computer/bo





http://peterhigginson.co.uk/LMC/



In Peter higginsons LMC the code is slightly different but does the same thing.

Inp

Sta 15

Inp

Sta 16

Inp

Sta 17

Inp

Sta 18

Inp

Sto 19

Lda 15

Add 16

Add 17

Sub 18

Sub 19

Out

hlt

You can also do loops in Assembly Language using Little Man Computers.

To do:

Value= INPUT

Do while value >=0

print value

value -

next

end.

Can be represented in LMC as:

IN

STA 25

SUB ONE

BRZ 05

BRA 00

OUT

HLT

ONE DAT 01

Extension activity.

00 In

01 Sta a

02 In

03 Sta b

04 Lda b

05 Brz end

06 Sub one

07 Sta b

08 Lda count

09 Add a

10 Sta count

11 Bra start

12 Lda count

13 out

14 Hlt

15 Dat 01

16 Dat 02

17 Dat 03

18 Dat 04

**Fibonacci Code**

Add the previous two numbers together.

1 1 2 3 5 8 13 21 34 55 89 144

**Teaching Ideas.**

Show a Powerpoint having some fun identifying terms like Date a Bus = Databus.

Important figures in history of computing architecture research project.

Von Neumann

Babbage

IBM creator Hermann Hollerich

Alan Turing.

**Data Compression.**

Data can be reduced in size to save space this is called compression.

You can use software like Zip software to do this for you but the principle is the same.

For example compression can be achieved by converting and reducing text to prevent repeated words.

So for example:

Pitter Patter

Pitter Patter

Listen to the Rain

Pitter Patter

Pitter Patter

On the Window Pain.

Becomes:

Pit Pat

Listen to the rain

On window pain.

**Teaching ideas.**

csunplugged.co.uk