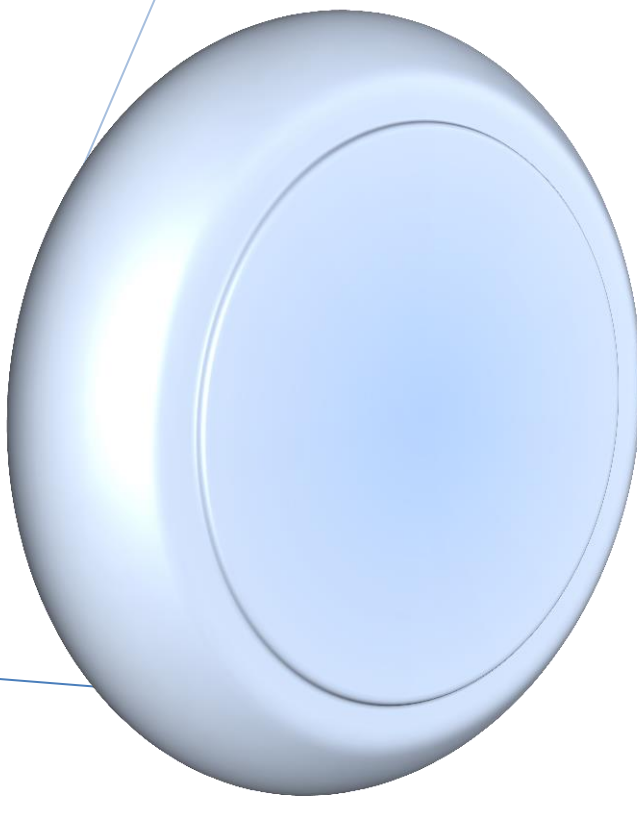
COMP 4 Coursework

Fluid dynamics simulator

This is the AQA COMP 4 coursework. My project is on building a fluid dynamics simulator for my client. In this document you will find the analysis, design, implementation, system maintenance and testing stages of the systems life cycle as well as the appendix.

Victor Sim
3/26/2015



Contents

I will be using a lot of physics terms throughout my project so I thought it would be a good idea to have a list of key terminology. Here it is as follows:

Key Terminology

Coefficient of viscosity (η) (Pas) – The viscosity of the fluid which indicates how viscous it is.

Density (ρ) (kgm^{-3}) – How much the object weights per unit volume.

Laminar flow – A type of flow where the flow lines are parallel to each other and do not intersect. The velocity at any point of the fluid surrounding the ball is the same.

Turbulent flow – A type of flow where the flow lines are unpredictable and intersect each other. The velocity at different points are different.

Viscous drag (N) – The resistive force applied on an object moving through a fluid with laminar flow around it.

Up-thrust (N) – The force acting up against an object falling through the fluid.

Terminal velocity (ms^{-1}) – The constant velocity that a freely falling object eventually reaches when the resistance of the medium through which it is falling prevents further acceleration. It is also the highest velocity attainable by an object in free fall.

g – The gravitational field strength constant which has a value of $9.81ms^{-2}$

SUVAT equations

S – Displacement (m)

U – Initial velocity (ms^{-1})

V – Final velocity (ms^{-1})

A – Acceleration (ms^{-2})

T – Time (s)

Equations

$F=6\pi r\eta v$ where r is the radius of the sphere, v is the velocity of the sphere and η is the coefficient of viscosity of the fluid.

Terminal velocity = $(2g(\rho_s - \rho_f)r^2)/9\eta$

Up-thrust = ρ_f x $Volume_{sphere}$

Volume of sphere = $4\pi r^3/3$

Weight of the ball = volume of sphere * density of ball

Up-thrust = volume of sphere * density of fluid

Analysis

Fluid dynamics simulator

In this section of the coursework, I will be talking about how I identified the problem of the current system and what my client wants for the new system. I will briefly talk about the data dictionary of the system and will go deeper into this with validation checks etc. in the design section. I will also discuss the different potential solutions and which method is the best way to implement the solution.

Victor Sim
3/26/2015

1. Analysis

1.1 The problem: identification and background

The organisation I am working for is Tanglin Trust School. It is in particular the physics department and it has 5 teaching staff. They teach students and my main contact is Mrs.Ridley who is one of the physics teachers. The students are learning the Edexcel AS physics syllabus and the teachers are teaching it. One of the topics in the syllabus is on fluid dynamics involving things like Stokes' law and up thrust. In the alternative to practical paper, they might ask about how to carry out an experiment to work out the terminal velocity of an object in a fluid. The problem is that some students aren't getting the concept and although they can do the experiment it can only be done at school and they might want to visualize how it's done at home. The students may also want to experiment more using different types of fluids or different size balls but are short of time in class. This was found out through interview 1 (which can be found in the appendix). They also aren't able to see the different values of the velocity, viscous drag and so on when the ball is falling down the tube. By doing the current practical experiment, the users will only know what the terminal velocity of the ball is in a particular fluid. I found out about the current practical experiment through observations. A picture of a typical set up can be found below.

1.2 Description of the current system

Currently, the teachers will teach the students on the whiteboard or an interactive screen for the students to look at. The students will copy down the notes and keep in their folders for future reference. The students can go back and revise out of their textbooks or look at their notes. During class time, there are also some practical sessions where we carry out a practical. This usually takes up a whole class to set up and carry it out. The experiment I am interested in is the fluid dynamics experiment where they try to work out the terminal velocity. This experiment in detail goes like this.

Equipment involved:

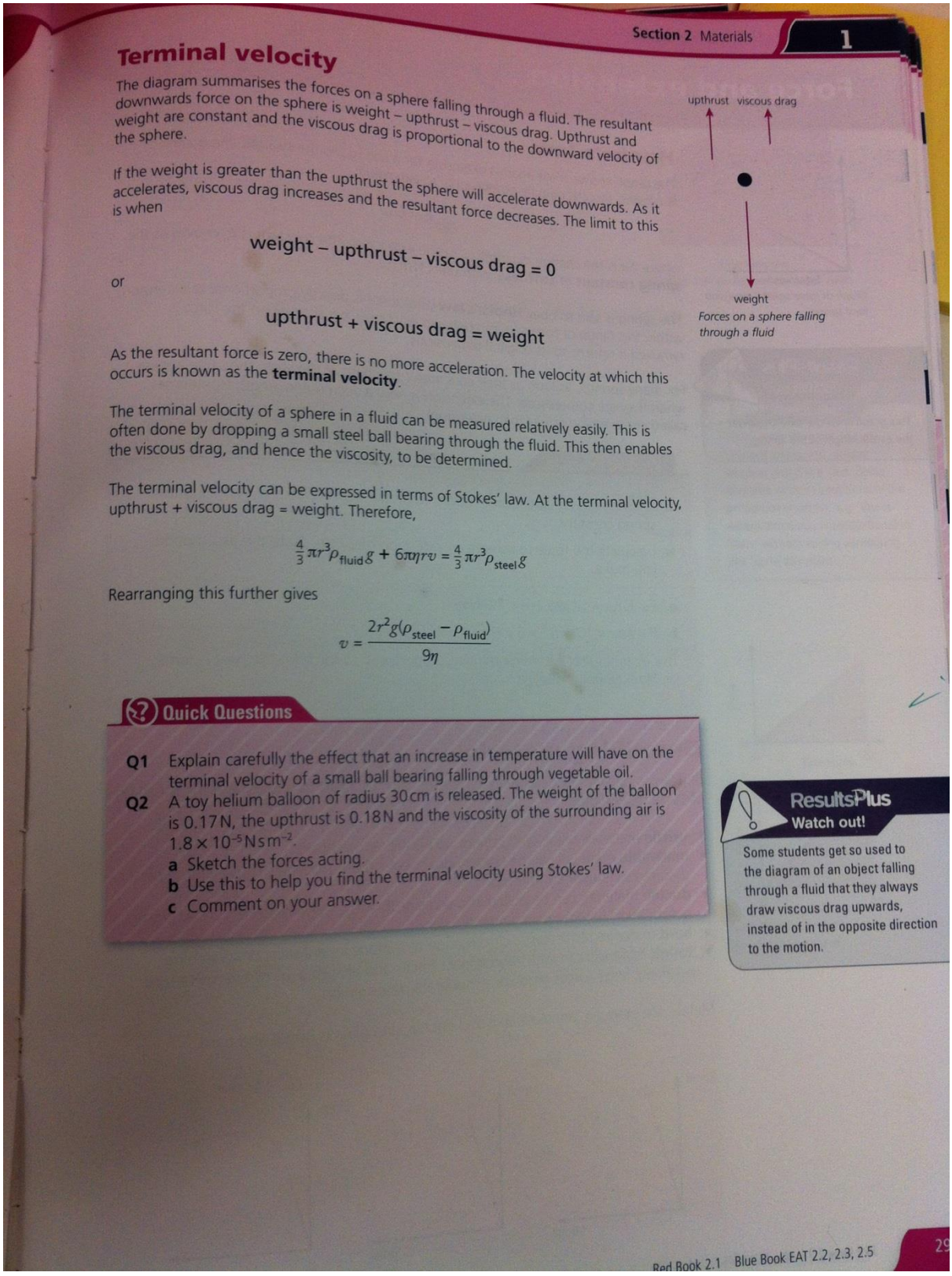
- Light gates
- Tube
- Fluid (e.g. glycerol)
- Ball (e.g. 8mm steel ball)
- Magnet (for getting the ball out of the tube)
- Clamps
- Retort stand

First attach the clamp to the retort stand then clamp the tube. Fill the tube with the fluid to be used. Use another retort stand to clamp two light gates a good distance apart. Also, make sure the light gates are low down the tube enough so that the ball is moving at terminal velocity as it passes through them. Attach the data logger to the light gates and finally drop the ball inside the tube, starting the data logger as you do so. You have to be careful when dropping the ball inside the tube as if it's too near the side it will produce turbulent flow which we don't want. After the run has finished, you can see the time it takes to pass through each light gate. Divide the diameter of the ball by the time to get the velocity which the ball was moving through. The results you get for both light gates should be similar and you can take an average to get the average terminal velocity. If they are

not the same, the velocity at the upper light gate will probably be slower because it hasn't reached terminal velocity yet. In this case the light gate will have to be moved lower down.

A typical set up would be something like this:

An example of a textbook page would be something like this:



Section 2 Materials 1

Terminal velocity

The diagram summarises the forces on a sphere falling through a fluid. The resultant downwards force on the sphere is weight – upthrust – viscous drag. Upthrust and weight are constant and the viscous drag is proportional to the downward velocity of the sphere.

upthrust viscous drag

weight

Forces on a sphere falling through a fluid

If the weight is greater than the upthrust the sphere will accelerate downwards. As it accelerates, viscous drag increases and the resultant force decreases. The limit to this is when

$$\text{weight} - \text{upthrust} - \text{viscous drag} = 0$$

or

$$\text{upthrust} + \text{viscous drag} = \text{weight}$$

As the resultant force is zero, there is no more acceleration. The velocity at which this occurs is known as the **terminal velocity**.

The terminal velocity of a sphere in a fluid can be measured relatively easily. This is often done by dropping a small steel ball bearing through the fluid. This then enables the viscous drag, and hence the viscosity, to be determined.

The terminal velocity can be expressed in terms of Stokes' law. At the terminal velocity, upthrust + viscous drag = weight. Therefore,

$$\frac{4}{3}\pi r^3 \rho_{\text{fluid}} g + 6\pi\eta r v = \frac{4}{3}\pi r^3 \rho_{\text{steel}} g$$

Rearranging this further gives

$$v = \frac{2r^2 g(\rho_{\text{steel}} - \rho_{\text{fluid}})}{9\eta}$$

Quick Questions

Q1 Explain carefully the effect that an increase in temperature will have on the terminal velocity of a small ball bearing falling through vegetable oil.

Q2 A toy helium balloon of radius 30 cm is released. The weight of the balloon is 0.17 N, the upthrust is 0.18 N and the viscosity of the surrounding air is $1.8 \times 10^{-5}\,\text{N s m}^{-2}$.

a Sketch the forces acting.

b Use this to help you find the terminal velocity using Stokes' law.

c Comment on your answer.

ResultsPlus
Watch out!

Some students get so used to the diagram of an object falling through a fluid that they always draw viscous drag upwards, instead of in the opposite direction to the motion.

Red Book 2.1 Blue Book EAT 2.2, 2.3, 2.5

Other examples of textbook pages about this topic:

Terminal velocity

You have previously learned that the acceleration due to gravity, near the surface of the Earth, is $9.81\,m\,s^{-2}$. An object falling in a vacuum does indeed accelerate at this rate. However, it is unusual for objects to be dropped near the surface of the Earth in a vacuum (in nearly all such cases a physics teacher is likely to be demonstrating to a class!). In reality, in order to calculate an object's actual acceleration when falling, we need to go back to Newton's second law. We know that $a = \Sigma F/m$. If we can take account of all the forces acting on an object, and combine these to find a resultant force, we can calculate the resulting acceleration.

fig. 2.1.14 A skydiver will fall at a constant speed if the forces acting on them are balanced.

For a falling object like a skydiver, we need to include the weight, the **upthrust** caused by the object being in the fluid air, and the **viscous drag** force caused by the movement. The tricky part is that the viscous drag varies with speed through the fluid, and that is constantly changing as a result of the acceleration. Usually, we consider the equilibrium situation, in which the weight exactly balances the sum of upthrust and drag, meaning that the falling velocity remains constant. This constant velocity is the **terminal velocity**.

Viscous drag

You would find it difficult to wade through a swimming pool filled with treacle because of the treacle's viscous drag. This is the friction force between a solid and a fluid. Calculating this fluid friction force can be relatively simple. On the other hand, it can be very complicated for large objects, fast objects, and irregularly shaped objects, as the turbulent flow creates an unpredictable situation.

HSW Stokes' Law

In the mid-nineteenth century, Sir George Gabriel Stokes, an Irish mathematician and physicist at Cambridge University, investigated fluid dynamics and derived an equation for the viscous drag (F) on a small sphere at low speeds. This formula is now called Stokes' Law:

$$F = 6\pi r \eta v$$

where r is the radius of sphere (m), v is the velocity of sphere ($m\,s^{-1}$), and η is the coefficient of viscosity of the fluid (Pa s).

Thus in such a simple situation, the drag force is directly proportional to the radius of the sphere, and directly proportional to the velocity, neither of which is necessarily an obvious outcome.

Stokes' publication of this law was delayed slightly while he considered the news that similar conclusions had previously been made by scientists in other parts of Europe, notably Navier and Poisson. At that time, communication between scientists was much slower and more limited than it is now, and it was common for the same results to be discovered independently and simultaneously. In this case, Stokes decided that his work was sufficiently different from that of the others to justify publishing it.

For simplicity, we will only consider simple situations, like a solid sphere moving slowly in a fluid. Imagine a ball bearing dropping through a column of oil, for example.

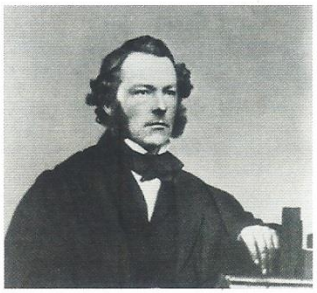


fig. 2.1.15 Along with Lord Kelvin and James Clerk Maxwell, Sir George Gabriel Stokes helped to build the reputation of Cambridge University in many areas of mathematical physics.

If you consider the terminal velocity of the ball bearing in terms of the forces in detail, then:

weight = upthrust + Stokes force

$m_s g$ = weight of fluid displaced + $6\pi r\eta v_{term}$

where m_s is the mass of the sphere and v_{term} is its terminal velocity.

For the sphere, the mass m_s is given by:

m_s = volume × density of sphere = $\frac{4}{3}\pi r^3 \times \rho_s$

so the weight of the sphere W_s is given by:

$W_s = m_s g = \frac{4}{3}\pi r^3 \rho_s g$

For the sphere, the upthrust is equal to the weight of fluid displaced. The mass m_f of fluid displaced is given by:

m_f = volume × density of fluid = $\frac{4}{3}\pi r^3 \times \rho_f$

so the weight of fluid displaced W_f is given by:

$W_f = m_f g = \frac{4}{3}\pi r^3 \rho_f g$

Overall then:

$$\frac{4}{3}\pi r^3 \rho_s g = \frac{4}{3}\pi r^3 \rho_f g + 6\pi r\eta v_{term}$$

We can rearrange the equation to find the terminal velocity:

$$v_{term} = \frac{\frac{4}{3}\pi r^3 g\,(\rho_s - \rho_f)}{6\pi r\eta}$$

Cancelling the π and the radius term:

$$v_{term} = \frac{2r^2 g\,(\rho_s - \rho_f)}{9\eta}$$

So terminal velocity is proportional to the square of the radius. This means that a larger sphere falls faster. And because the radius is squared, it falls much faster! Don't forget that this is based on a slow-moving small sphere – more complex situations have more complex equations.

It must be remembered that the simple slow-falling sphere is not a common situation and in most real applications the terminal velocity value is a result of more complex calculations. However, the principle that larger objects generally fall faster holds true for most objects without a parachute.

Falling object	Terminal velocity/m s^{-1}
Skydiver	60
Golf ball	32
Hail stone (0.5 cm radius)	14
Raindrop (0.2 cm radius)	9

table 2.1.5 Some typical real terminal velocities in air.

Worked examples

To work out the terminal velocity of two different sizes of steel ball bearing falling through glycerine in a measuring cylinder, we need to know the densities of steel and glycerine, along with the viscosity of glycerine and the radii of the two ball bearings.

The viscosity of glycerine is highly temperature dependent: at 20 °C we can take η = 1.5 Pa s

Density of steel = 7800 kg m^{-3}

Density of glycerine = 1200 kg m^{-3}

g = 9.81 m s^{-2}

a For a 1 mm radius ball bearing:

$$v_{term} = \frac{2r^2 g\,(\rho_s - \rho_f)}{9\eta}$$

$$v_{term} = \frac{2\,(1\times10^{-2})^2 \times 9.81 \times (7800 - 1200)}{9 \times 1.5}$$

$$= 9.6 \times 10^{-3}\ \text{m s}^{-1}$$

b For a 2 mm radius ball bearing:

$$v_{term} = \frac{2r^2 g\,(\rho_s - \rho_f)}{9\eta}$$

$$v_{term} = \frac{2\,(2 \times 10^{-3})^2 \times 9.81 \times (7800 - 1200)}{9 \times 1.5}$$

$$= 3.8 \times 10^{-2}\ \text{m s}^{-1}$$

Comparing the values, we see that doubling the ball's radius makes its terminal velocity four times as great.

Questions

1 Use Stokes' Law to calculate the viscous drag on a ball bearing with a radius of 1 mm, falling at 1 mm s^{-1} through liquid chocolate at 30 °C.

2 Why is it difficult to calculate the terminal velocity for a cat falling from a high rooftop?

3 A spherical meteorite, of radius 2 m and made of pure iron, falls towards Earth.

- **a** For its fall through the air, calculate the meteorite's terminal velocity.
- **b** It lands in a tropical freshwater lake at 20 °C and continues sinking underwater. Calculate its new terminal velocity.
- **c** What assumptions have you made in order to make these calculations?

(See table 2.1.1 for density data and table 2.1.4 for viscosity data.)

Here is the topic area I am mainly dealing with in the specification:

Context approach

Students will be assessed on their ability to:	Suggested experiments
18 understand and use the terms *density*, *laminar flow*, *streamline flow*, *terminal velocity*, *turbulent flow*, *upthrust* and *viscous drag*, for example, in transport design or in manufacturing	
19 recall, and use primary or secondary data to show that the rate of flow of a fluid is related to its viscosity	
20 recognise and use the expression for Stokes's Law, $F = 6\pi\eta rv$ and upthrust = weight of fluid displaced	
21 investigate, using primary or secondary data, and recall that the viscosities of most fluids change with temperature. Explain the importance of this for industrial applications	

1.3 Identification of the prospective user(s)

This program is for the physics department of Tanglin Trust School to use but the final users would be the students of the school who are taking physics. About 30 people (depending on how many people are taking the subject) will use the system. The role of the students is to use this program to help them in their learning. This program is a physics fluid dynamics simulator program where they can get the same data out which they would get by doing the experiment. They will be able to adjust the size and density of the ball to see how this affects the terminal velocity through a fluid.

1.4 Skills of user

The user does not require any special computer knowledge to use this software. The user only needs to know how to use a computer i.e. use a mouse and a keyboard. The user may need to know some physics terminology and basic maths skills to know how the software works and how to use the software. Since the users whom my program is intended for would be taking the AS physics course, they should have little trouble with this.

1.5 Identification of user needs and acceptable limitations

I've used interviews with my client, Mrs. Ridley a Physics teacher of Tanglin Trust School, as my main resource to collect information about what the objectives of my system. The interviews can be found in the appendix pages 108 – 111.

Sometimes it is hard for the students to picture what is going on so my program would make it easier to understand because they will be able to see how the data values changes as the experiment is going on. According to my client, the system needs to provide an accurate simulation of how the ball would move through a fluid. It also needs to display information on the screen about the ball's path (e.g. terminal velocity) and characteristics (e.g. weight) as well as information about the fluid (e.g. viscosity). The system needs to give the option of fluids: glycerine, sunflower oil, coconut oil and olive oil. They need to be able to choose different types of balls including a 6mm steel ball, 8mm steel ball, 6mm marble ball and an 8mm marble ball or able to create their own

custom ball with a specified radius and density. The system should also be able to allow the user to export the data from the simulation. This is useful as the user (the students) can experiment with the data they got to see how the variables affect one another. As part of the additional objectives, the system must provide a quiz mode where the system will display a good message if the user answered the question correctly and gives them a negative message if the user did not answer it correctly. This quiz mode will be able to test the user's understanding of the topic area.

A limitation the system has is that the user can't see the density of the steel or marble ball. This is because I found it a bit unnecessary to display the density of the ball in another label when you know your volume and weight. There is a label, however which tells you the density of the ball if you change it with the slider. If the user wishes to know the density of the ball without calculating it, they can export the data and it will state the value of density of the ball there. Another limitation of the program is that it assumes laminar flow for all types of ball and fluids. It doesn't actually check whether those values would produce turbulent flow or not. Another limitation of my program is the ball might not reach terminal velocity during the simulation for all values of the ball's radius and density. This is because the range of values for the velocities of the different balls in the different fluids are too great for the simulation to end in a short enough of time for a light ball and reach terminal velocity for a heavy ball. E.g. a ball 2mm radius, $2gcm^{-3}$ in glycerol fluid would take about 30 seconds real time to reach the bottom of the tube after reaching terminal velocity whereas a ball 8mm, $10gcm^{-3}$ in glycerol would take 2 seconds to reach the bottom of the tube without reaching terminal velocity. The client said during an interview (please refer to interview 3 in the appendix page 110) that she would much prefer for the students to have that range of inputs.

1.6 Constraints

Hardware limitations

The program would be burned onto a CD-R and given to my client so the user needs to be able to install the program from a CD-drive. It could be a hardware limitation if the user's laptop doesn't have a CD-drive. This could be overcome by loading the files onto a school computer which all have CD-drives and copying the files onto a thumbdrive for the student to have their own copy. The size of the files shouldn't be too large so should be able to fit on a thumbdrive with minimum storage capacity. The specifications of the school Dell computers are an Intel® Core™ i7-4770S CPU @ 3.1GHz and has a RAM of 8GB in case the students or teachers wish to install the program onto the school computer.

Software limitations

A software limitation is that this program would only work on the Windows operating system and won't work on a Mac operating system.

Time limitations

The program would have to be finished before the end of this academic year. This is so that the students taking the course now will have the opportunity to use the program. Also, the teacher could have some practice on how to utilise the program for next year's use e.g. using it during an experiment to check experimental values with theoretical values.

Knowledge limitations

The users (students) will have to have basic knowledge of the mathematical and physics term to be able to use this program so they know what variable they are changing e.g. radius and what the outputs mean e.g. terminal velocity. They will also need to know how to use a computer but no special knowledge which can't be found in the user manual is needed.

Access rights limitations

Optimally, only the teachers would have access rights to the question files which contains the questions and answers for the quiz. This can be done by encrypting the file which requires a password to open it. However, this caused some difficulties which is discussed more in the system security of the design section and in the appraisal section.

1.7 Methods of fact finding

Questionnaires

I gave some questionnaires to a sample of 5 students from the Physics class. I asked them to fill it out and send it back to me by e-mail. From the questionnaires, I was able to find out what they found most useful to be in the program and what they would like in the program so I could try and include them. The questionnaires (pages 103 – 107) can be found in the appendix.

Documentation

I used some textbook pages to see what the students had to learn. I tried incorporating all these ideas into my program so that all the key parts which they need to know can be found out from my program. E.g. how does the radius affect the terminal velocity? Samples of the textbook pages I used can be found above right after the description of the current system section. I have also used the specification which can be found on the Edexcel website for their course to know exactly what they need to cover and what my program needs. The specification related to my project can also be found in the previous pages (page 11).

Interviews

I got to ask what my client wants in detail during interviews. This helped me get a basic understanding of how to design my program during the first interview. I frequently went back and had interviews during the design and implementation process to ask any questions about what the client wants and showing her the work in progress to make sure everything is fine with her (interviews can be found in the appendix pages 108 – 111).

Observation

I got to know about the current system very well because I took the AS physics course last year. So I've experienced using the current system. I have sat in the classes this year and the procedure of teaching is still fairly the same. I think with the proposed system I could have experimented with the topic more and discovered new things. A picture of the apparatus I set up last year can be found on page 7 in the description of the current system section.

1.8 Data source(s) and destination(s)

Inputs from user:

- Which ball the user wants to experiment with. The user can select it from a drop down menu.
- What fluid the user wants to use. The user can select this from a drop down menu
- The user can change the diameter of the ball to create a custom ball using a slide bar
- The use can change the density of the ball to create a custom ball using a slider bar
- The user can select which mode they want by clicking on the buttons
- The user would answer the questions in a textbox
- The user is able to specify the location in which he/she wishes to output the file and the name he/she wishes to call it

Outputs:

- The current speed of the ball will be displayed in a label
- The current speed will stay the same when it reaches the terminal velocity
- The viscous drag will be displayed in a label
- The constants (for a particular ball and fluid):
    - Upthrust
    - Weight
    - Volume
    - Fluid density
    - Fluid viscosity
- will be displayed in labels
- The user would be able to see the ball and fluid she/he selected
- The ball would be loaded from a bitmap file
- The tube would be loaded from a bitmap file
- The bitmap files would be displayed on the screen in pictureboxes
- The ball and colour of the fluid will change accordingly
- The user would be able to export the data from the simulation
- Sound effects would be heard when buttons are clicked
- A new question will be outputted in quiz mode when an answer is inputted correctly
- A csv file containing the data from the simulation

1.9 Data volumes

Current system

The data currently stored in the current system are textbooks, teacher's notes, students' notes, worksheets and 'equipment'. The teacher would refer to his/her notes every lesson to teach the students and the student would make new notes every lesson. We have physics lessons 5 times a week so these data will increase with about 5 pages of new notes every week for the student. The textbook gets referred to at least once a week, either the teacher is using it in class to show a diagram from it or something or the student is reading it back at home. The equipment is used

roughly about once a month as practical lessons are not as common as content learning based lessons.

Proposed system

Since students want to try many experiments, you can expect there to be about 10 different outputs of terminal velocity. The data needed to process the terminal velocity would include things like the density of the fluid, the density of the solid, the radius, the viscosity of the fluid and the constant for gravitational field strength. The calculation to calculate the current speed and to move the ball by a distance would need to reiterate many times per run because it needs to update regularly using the tick timer. The timer would update at an interval of 1 and if the program is used about 10 times a day per student so it would do the calculation several hundreds of times.

The fluid dynamics simulator program can also output the data from the simulation. Depending on how long the simulation is, this file may be from 1 to 50kB.

Inputs

Custom ball – 63kB

Steel ball – 204kB

Marble ball – 134kB

Database of questions – 10kB

Initial test tube.jpg (with no colour) – 1kB

Glycerol, sunflower oil, coconut oil, olive oil test tubes (with colour) – 8kB

Button click sound – 59kB

Correct sound – 302kB

Incorrect sound – 523kB

Outputs

Simulation data – 50kB

1.10 Analysis data dictionary (from perspective of end user)

Outputs

Name	Description	Data type	Size/range
Viscous drag	Shows the instantaneous viscous drag acting on the ball	Double	
Current velocity	Shows the distance the ball should move by in a given time. The rate	Double	0-0.5ms^{-1}

	of change of displacement.		
Terminal velocity	When the maximum velocity is reached, the YVelocity becomes the TVelocity (terminal velocity)	Double	1 – 100 pixels per second 0-0.25ms^{-1}
Upthrust	Shows the upthrust on the ball	Double	0.0001 – 0.03 N
Weight	Shows the weight of the ball	Double	0.0005 – 0.2N
Volume	Shows the volume of the ball	Double	3.35x10^{-8}m^3 -
Fluid density	Shows the density of the fluid	Double	
Fluid viscosity	Shows the viscosity of the fluid	Double	
Time	The time elapsed since the ball was dropped in the fluid	Double	0-0.5 seconds
ExperimentNo.	Stores the number of the experiment	Integer	1-30
YDisplacement	Determines how much the ball should move down	Double	200-400 pixels

Inputs

Name	Description	Data type	Size/range
Timer.enabled	Is the timer enabled	Boolean	True/False
CoefficientOfViscosity	A constant used in the equation $v_s = \frac{2}{9}\frac{(\rho_p - \rho_f)}{\mu} g R^2$	Double	1 x 10^-3 Pas^{-1} -1.5 Pas^{-1}
Fluid	Stores the fluid currently being used	String	Glycerol, Sunflower oil, Coconut oil, Olive oil
Material	Stores the material of the ball currently being used	String	Steel, Marble
Radius	Gets the radius of the ball	Integer	2mm-8mm
Density	Gets the density of the ball	Double	2g/cm^3 – 10g/cm^3
Answer	The answer the user inputs into the textbox of the quiz	String	

Constants

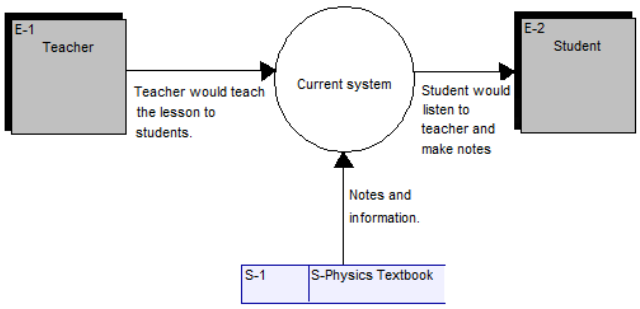
SteelDensity	A constant for the	Single	8.05gcm^{-3}

	density of steel		
MarbleDensity	A constant for the density of marble	Single	2.40 gcm^{-3}
GlycerolViscosity	A constant for the coefficient of glycerol	Single	1.412 Pas
SunFlowerOilViscosity	A constant for the viscosity of sunflower oil	Single	0.4914 Pas
CoconutOilViscosity	A constant for the viscosity of coconut oil	Single	0.06 Pas
OliveOilViscosity	A constant for the viscosity of Olive oil	Single	0.1075 Pas
GlycerolDensity	A constant for the density of glycerol	Single	1.26g/cm^3
SunFlowerOilDensity	A constant for the density of sunflower oil	Single	0.93g/cm^3
CoconutOilDensity	A constant for the density of coconut oil	Single	0.924g/cm^3
OliveOilDensity	A constant for the density of Olive oil	Single	0.86g/cm^3
g	A constant for the gravitational field strength of the Earth	Double	9.81N/kg

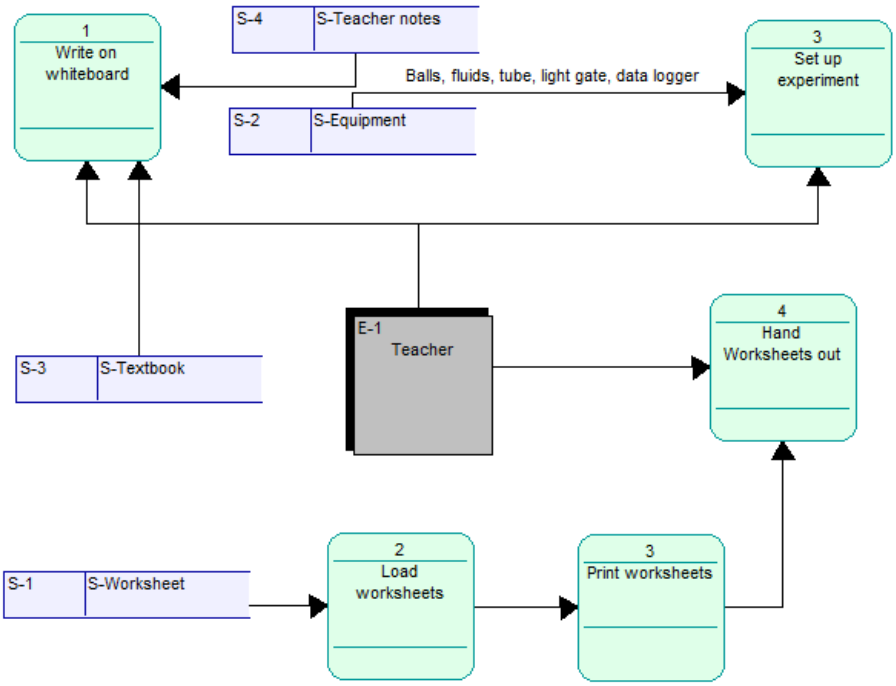
All the viscosities are the viscosities at room temperature

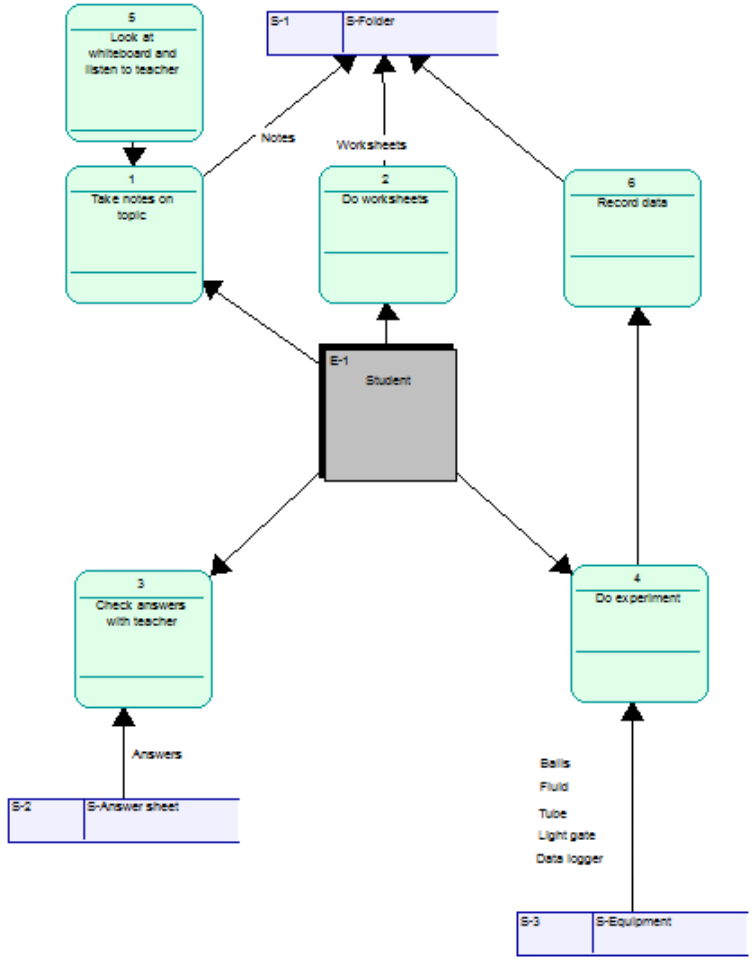
1.11 Data flow diagrams

1.11.1 Data flow diagram level 0 for existing system

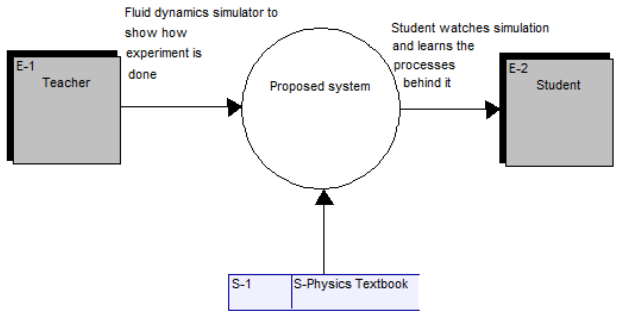


1.11.2 Data flow diagrams level 1 for existing system

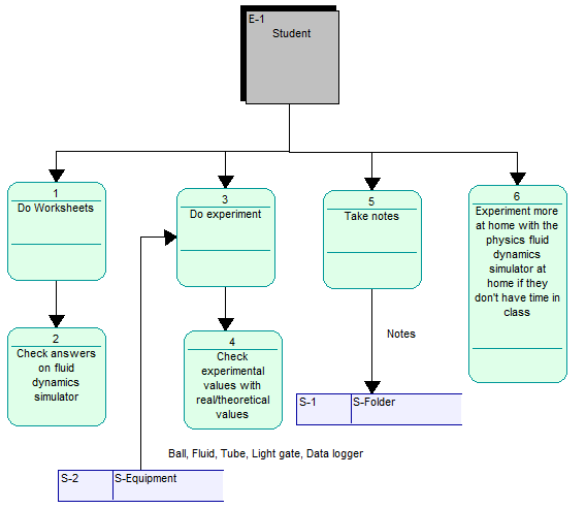


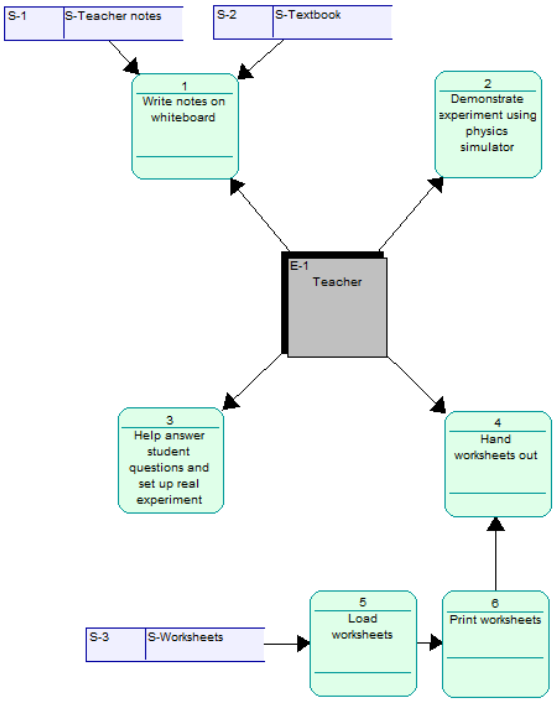


1.11.3 Data flow diagram level 0 for proposed system



1.11.4 Data flow diagrams level 1 for proposed system





1.12 Objectives for the proposed system

Original Objectives

1. It will make an animation of the ball in fluid
2. The ball will stop once it hits the bottom of the container
3. The user can change the fluid therefore changing the viscosity and density of the fluid
4. The user can change the ball i.e radius, density etc.
5. The user will be able to know the current speed
6. The user will be able to know the terminal velocity
7. The user will be able to know the density of the fluid
8. The user will be able to know the volume of the ball
9. The user will be able to know the up-thrust acting on the ball
10. The colour of the fluid would change as a new fluid is selected
11. The size of the ball would change as the radius property is changed
12. The picture of the ball would change if the material of the ball is changed (i.e. glass, steel or generic material which represents the density of the density slider bar)
13. Sound effects will be heard at the appropriate time
14. The time elapsed will be displayed
15. The user will be able to choose a fluid from glycerine, sunflower oil, coconut oil and olive oil
16. The user will be able to choose standard balls which are 8mm steel ball, 6mm steel ball, 8mm marble ball, 6mm marble ball
17. The animation of the ball would be smooth
18. The results will be outputted to 2 - 5 significant figures on the simulation
19. The data of the simulation would be outputted into a table and stored as an external file
20. The user can return to the main menu to switch between different modes

10.10.14.

Final Objectives

1. It will make an animation of the ball in fluid
2. The ball will stop once it hits the bottom of the container
3. The user can change the fluid therefore changing the viscosity and density of the fluid
4. The user can change the ball i.e radius, density etc.
5. The user will be able to know the current speed
6. The user will be able to know the terminal velocity
7. The user will be able to know the density of the fluid
8. The user will be able to know the volume of the ball
9. The user will be able to know the up-thrust acting on the ball
10. The colour of the fluid would change as a new fluid is selected
11. The size of the ball would change as the radius property is changed
12. The picture of the ball would change if the material of the ball is changed (i.e. glass, steel or generic material which represents the density of the density slider bar)
13. Sound effects will be heard at the appropriate time
14. The time elapsed will be displayed
15. The user will be able to choose a fluid from glycerine, sunflower oil, coconut oil and olive oil
16. The user will be able to choose standard balls which are 8mm steel ball, 6mm steel ball, 8mm marble ball, 6mm marble ball
17. The animation of the ball would be smooth
18. The results will be outputted to 2 - 5 significant figures on the simulation
19. The data of the simulation would be outputted into a table and stored as an external file
20. The user can return to the main menu to switch between different modes
21. The user can play a quiz mode
22. A positive result will be returned when the user gives the correct answer in quiz mode
23. A negative result will be returned when the user gives a wrong answer in quiz mode
24. The system will keep track of the user's score based on how many he/she got right in quiz mode

After some discussion with my client, we decided to add additional features (the 4 last objectives) which may be useful for the students' use.

Some things I discussed with my client but never really got made into an objective because they weren't essential are:

- There will be fluid lines around the ball to show the type of flow (laminar flow)
- The system would limit the user's inputs if it would cause there to be turbulent flow
- The distance it travelled till it reached terminal velocity will be displayed

1.13 Realistic appraisal of the feasibility of potential solutions

Manual solutions

The current system is a manual system. This means that all experiments must be carried out by hand. The students might not know how to use some of the equipment such as the data logger. The student can only use the equipment that is available to them in the laboratory and they might have to share the equipment with different classes so it might be unavailable.

Advantages	Disadvantages
The students can get a better idea of how to write up the practical in the exam	It takes valuable class time away
The students will learn how to use the equipment used in the experiment	The equipment might not be functioning properly
Students will recognise the disadvantages and advantages of using different equipment	There will be some practical errors involved such as turbulent flow created when the ball is too near the side of the tube
	The equipment might not be available when needed
	The student would need to wait for the teacher to mark the answers
	Students will have to compare their results with other groups to see if they got similar answers to get an accurate average

Off-the shelf software solutions

The website http://www.falstad.com/mathphysics.html offers many physics programs but they currently do not offer a Physics fluid dynamics simulator program.

Advantages	Disadvantages
It is already there so they can use it straight away and don't have to wait for development time	They might not provide the program
It has most of the general features needed for a fluid dynamics simulator	There might be unnecessary features of the program which makes it hard to use

Bespoke

I am now creating a bespoke solution to the current problem. This allows the programmer to work directly with the client so the programmer is able to add all the necessary features and possibly bonus features.

Advantages	Disadvantages
It meets all the specific needs of the end user	It takes time to develop it
The programmer can make it as easy to use for the user by asking them what they think about the user interface	It has not been tested by many users so may contain some bugs
Students are able to use their creativity to explore what would happen in different situations	
The students are able to answer questions and the program would be able to mark them straight away instead of waiting for a teacher	
The students will be able to see the difference between experimental values and actual values	

1.14 Justification of chosen solution (use of formal methods, e.g. observation, analysis of existing paperwork, interviews surveys)

I had interviews with my client and she told me many things (Please refer to appendix for interviews pages 108 – 111). From this I was able to know what she wants for the new system and what the problems with the current one are.

Possible solution 1 (manual solution)

One of the possible solutions was to keep the manual system. This means that all experiments are done manually and they would take time to try another fluid or switch to a different ball. An experiment can easily take up a whole lesson. This however, would mean that they can have a better understanding of how to use the equipment. The students will have to compare results with the other students to see if their results are right.

Possible solution 2 (bespoke software solution)

Another possible solution is to just purely use the physics fluid dynamics simulator to gain experimental data from it. This would save lots of class time as the user can just get it up and running very quickly on their computer and input the values they wish to put to see the result. Changing fluids or balls would be very quick using this as well as they can just select it from a drop down menu. They won't however have the practical aspects of this like how to set the equipment up which is important for the Paper 3 exam.

Possible solution 3(manual with bespoke software solution)

One more possible solution is to incorporate the bespoke software into the current manual system. This means the students would be able to have the practical knowledge of it like how to set the

experiment up and also be able to check their values quickly and accurately using the program. You can also change the variables more easily using my program than in real life. The book only has still pictures in the current system, whereas my one will have animations.

After the interview, I have chosen to use solution 3 because it takes the good things from both the current and proposed system.

Possible programming languages

C#

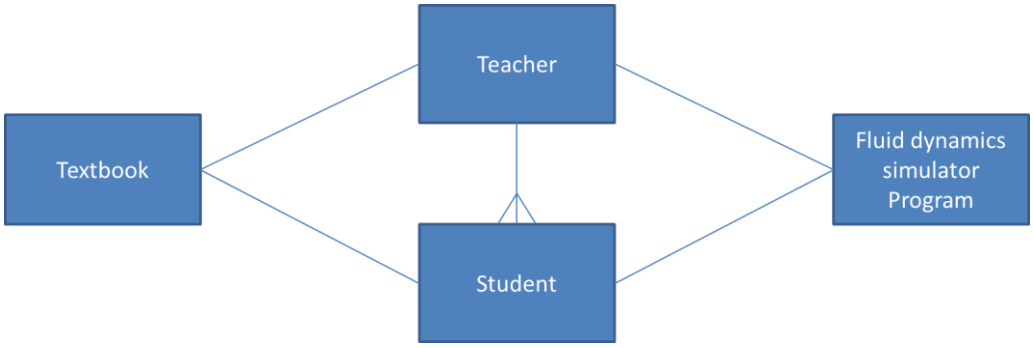
Advantages	Disadvantages
It can be used with OpenGL and Unity to create 3D images and make the program more visually interesting	Inexperience with language
It is a more powerful language	
Active forums to seek help from	

Visual Basic

Advantages	Disadvantages
I am already familiar with the language so I have more experience in using it and won't have as much trouble with the syntax as in the other languages	It doesn't support 3D graphics
Fairly simple to use	
Active forums to seek help from	

In the end I decided to use Visual Basic because I was already familiar with the language and didn't have to learn a new one. This would save time as I don't have to learn a brand new syntax. Both support object orientated programming and both have good online support. Visual Basic also meets all my essential needs so I am not limited to meet an important objective. They have a visual windows form application which I can work on so you can actually see where the object will be on the screen and is a nice working environment.

1.15 Entity – relationship (E-R) model and diagrams



1.16 Identification of objects and object – analysis diagrams for object-orientated programmed solutions

Ball class

Fields:

Access type	Field Name	Field Type	Initial value	Description
Private	p_radius	Integer	2mm	Stores the radius of the ball
Private	p_rensity	Double	$2gcm^{-3}$	Stores the density of the ball
Private	p_top	Double	210	Stores the vertical position of the ball
Private	p_material	String	Steel	Stores the material of the ball
Private	p_volume	Double	$3.35x10^{-8}m^3$	Stores the volume of the ball
Private	p_mass	Double	$6.7x10^{-5}kg$	Stores the mass of the ball
Private	p_weight	Double	$6.57x10^{-4}N$	Stores the weight of the ball
Private	p_image	PictureBox	Ball	Gives all the properties of the picturebox class to this variable

Methods:

Method name	Subroutine/Function	Parameters	Description
ChangeColour	Subroutine	Fluid	Changes the backcolour of the ball to match the backcolour of the fluid
CalculateVolume	Subroutine	Radius	Changes the volume field of the ball
CalculateMass	Subroutine	Volume, density	Calculates the mass of the ball from the volume and density
GetWeight	Function	None	Calculates and returns the weight of the ball.
GetDensity	Function	None	Uses if statement to see which density value to return and returns the density value

Fluid class

Fields:

Access type	Field Name	Field Type	Initial value	Description
Private	p_fluid	String	Nothing	Stores the fluid currently in use
Private	p_density	Double	Nothing	Stores the density of the fluid corresponding to the fluid
Private	p_viscosity	Double	Nothing	Stores the viscosity corresponding to the fluid currently in use
Private	p_image	PictureBox	Tube picture box displayed on form	Stores the properties of the picturebox on form into a field in the class

Methods:

Method name	Subroutine/Function	Parameters	Description
ChangeColour	Subroutine	None	Changes the colour of the fluid according to the fluid selected
GetDensity	Function	None	Returns the density of the fluid
GetViscosity	Function	None	Returns the viscosity of the fluid

Simulator class

Fields:

Access type	Field Name	Field Type	Initial value	Description
Private	p_oldVelocity	Double	0	Stores the previous velocity for calculations
Private	p_currentVelocity	Double	None	Stores the current velocity to be outputted to the user
Private	p_terminalVelocity	Double	None	Stores the value of the terminal velocity

Private	p_viscousDrag	Double	None	Stores the value of the viscous drag acting on the ball
Private	p_upthrust	Double	None	Stores the upthrust acting on the ball
Private	p_resultant	Double	None	Stores the value of the resultant force acting on the ball
Private	p_acceleration	Double	None	Stores the value of the acceleration of the ball

Methods:

Method name	Subroutine/Function	Parameters	Description
CalculateUpthrust	Subroutine	Volume, densityOfFluid	Calculates the value of upthrust by using volume*densityOfFluid*g*1000
CalculateAcceleration	Subroutine	Weight, mass	First calculates the resultant force by using weight – (viscous drag + up-thrust) and then using Newton's second law F=ma to calculate the acceleration
CalculateNewVelocity	Subroutine	None	Calculates the new velocity by using SUVAT. It assumes constant velocity for 0.001 seconds. Then updates the old velocity to be the current velocity for the next run
CalculateDistance	Subroutine	None	Calculates the distance travelled by multiplying the current velocity by the time (assuming constant velocity for the 0.001 s time interval) and multiplying it by 500000 to produce noticeable changes in the ball's movement on screen
CalculateTerminalVelocity	Subroutine	Radius, solid density, fluid density, viscosity	Calculates the terminal velocity by using the equation F= 6πrηv
CalculateViscousDrag	Subroutine	Radius, viscosity	Calculates the viscous drag acting on the ball
FluidChosen	Function	Fluid	Checks if a fluid is chosen or

			not. Returns a true/false result

Quiz Class

Fields:

Access type	Field Name	Field Type	Initial value	Description
Private	Score	Integer	0	Stores the current score of the user

Methods:

Method name	Subroutine/Function	Parameters	Description
readFile	Subroutine	Quiztype	Reads in the appropriate file containing the right set of questions and answers based on the quiztype parameter passed to it
CheckAnswer	Subroutine	Answer, byRef Correct	Checks the answer of the user. Converts the user's input to upper case so the answer is not case sensitive. Plays wav file and updates score if user's answer is correct. If not, plays a different wav file and outputs their total score with an appropriate message according to the score they obtained. Says "Need to work harder!" if score is 0, says "Good effort!" if score > 0 and says "Well done!" if score > 50
UpdateArray	Subroutine	None	Deletes the question which was just displayed to the user and updates the array. Redims the array to save space
GenerateQuestion	Function	None	Selects a random question using the rnd

			function. Extracts the question part of the slot in the array using the mid function. Returns the question. If the user has answered all questions correctly, then outputs a well done message followed by their score and returns nothing.

Originally, there was a LoadBitmap function in my program which goes like this:

Ball class

LoadBitmap	Function	Filename	Loads the bitmap file of the ball so the user can see it on screen, with a try catch statement in case it can't read the file name. Returns the bitmap.

Fluid class

LoadBitmap	Function	Filename	Loads the bitmap of the tube with the correct fluid (colour) in it. Returns the bitmap of the tube.

The code was like this:

```
Public Function LoadBitmap(ByVal filename As String)
        Dim bmp As Bitmap
        Try
            bmp = New Bitmap(filename)
        Catch ex As Exception
            bmp = Nothing
        End Try
        Return bmp
End Function
```

But they were taken out because this function only works when it's reading in a file locally. I don't want the end users to have access to the pictures so they can't change it or delete it by accident. Therefore, I stored the pictures in my resources in VB so they are saved in the executable file when the program is compiled.

Originally there was also a sound class but for the same reason as above, it got taken out. It is also saved in my resources and I just create an audio device of class Microsoft.VisualBasic.Devices.Audio in the forms where I use it to play the sound.

The original sound class went like this:

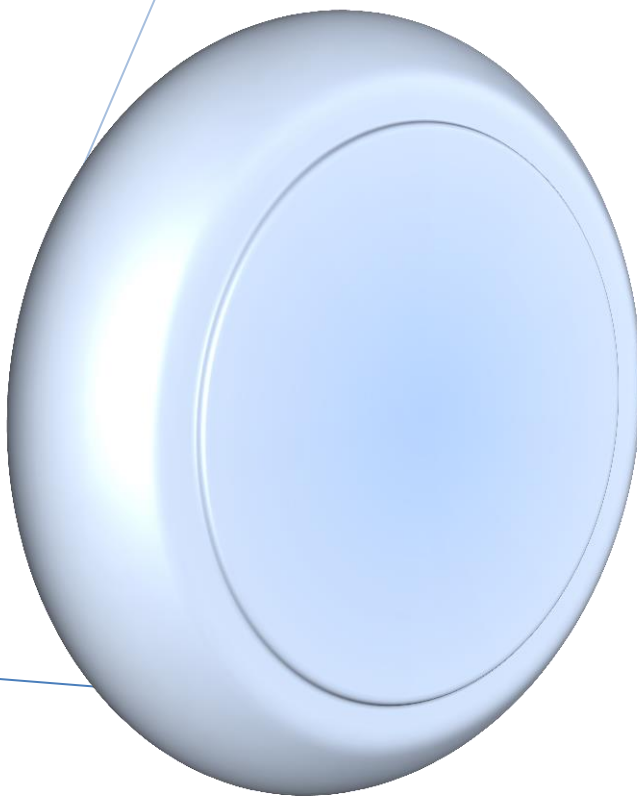
Sounds Class

Methods:

Method name	Subroutine/Function	Parameters	Description
PlayWavFile	Subroutine	Filename	Plays the wave files for sounds like clicking a button or getting a question right or wrong
LoadWave	Function	Filename	Tries to put soundlocation as the filename location in the parameter using a try catch statement. Returns the sound if successful else returns nothing

The code for the Sounds Class was like this:

```
Imports System.Media
Imports Microsoft.VisualBasic

Public Class Sounds
    Dim audio As System.Media.SoundPlayer
    Dim audioDevice As Microsoft.VisualBasic.Devices.Audio

    Public Sub PlayWavFile(ByVal filename As String)
        Try
            'load audio using the LowadWave function
            audio = LoadWave(filename)
        Catch ex As Exception
        End Try
        'create audio device
        audioDevice = New Devices.Audio()
        audio.Play()
    End Sub

    Public Function LoadWave(ByVal filename As String) As System.Media.SoundPlayer
        Dim sound As SoundPlayer = Nothing
        Try
            sound = New SoundPlayer()
            sound.SoundLocation = filename
            sound.Load()
        Catch ex As Exception
        End Try
        Return sound
    End Function
End Class
```



Design

Fluid dynamics simulator

This is the design section of the coursework. In this section, I will be looking at how to design my system with system flowcharts, record of database structure including validation checks, the design of the user interface and security and integrity of data.

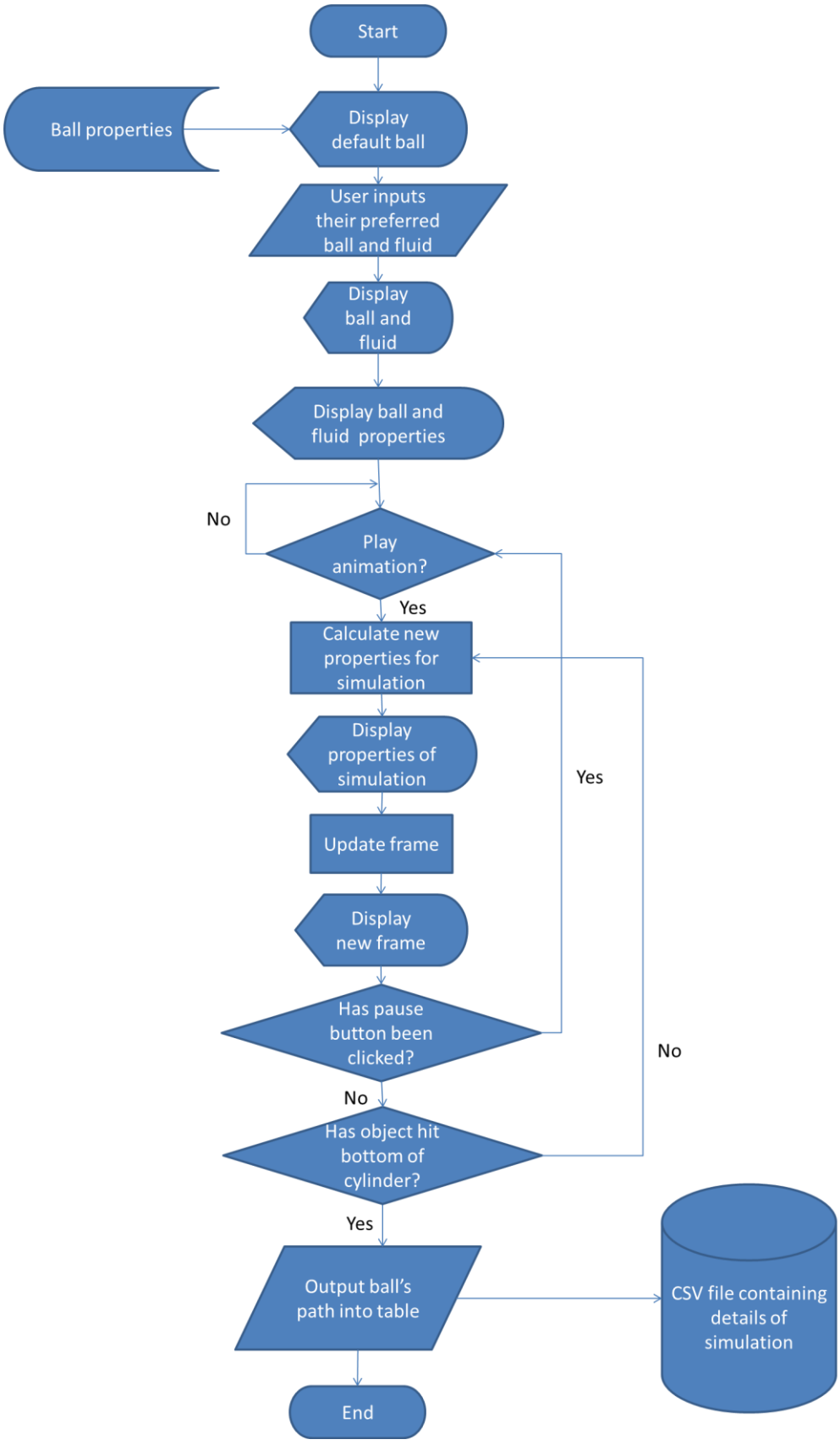
Victor Sim
3/26/2015

2. Design

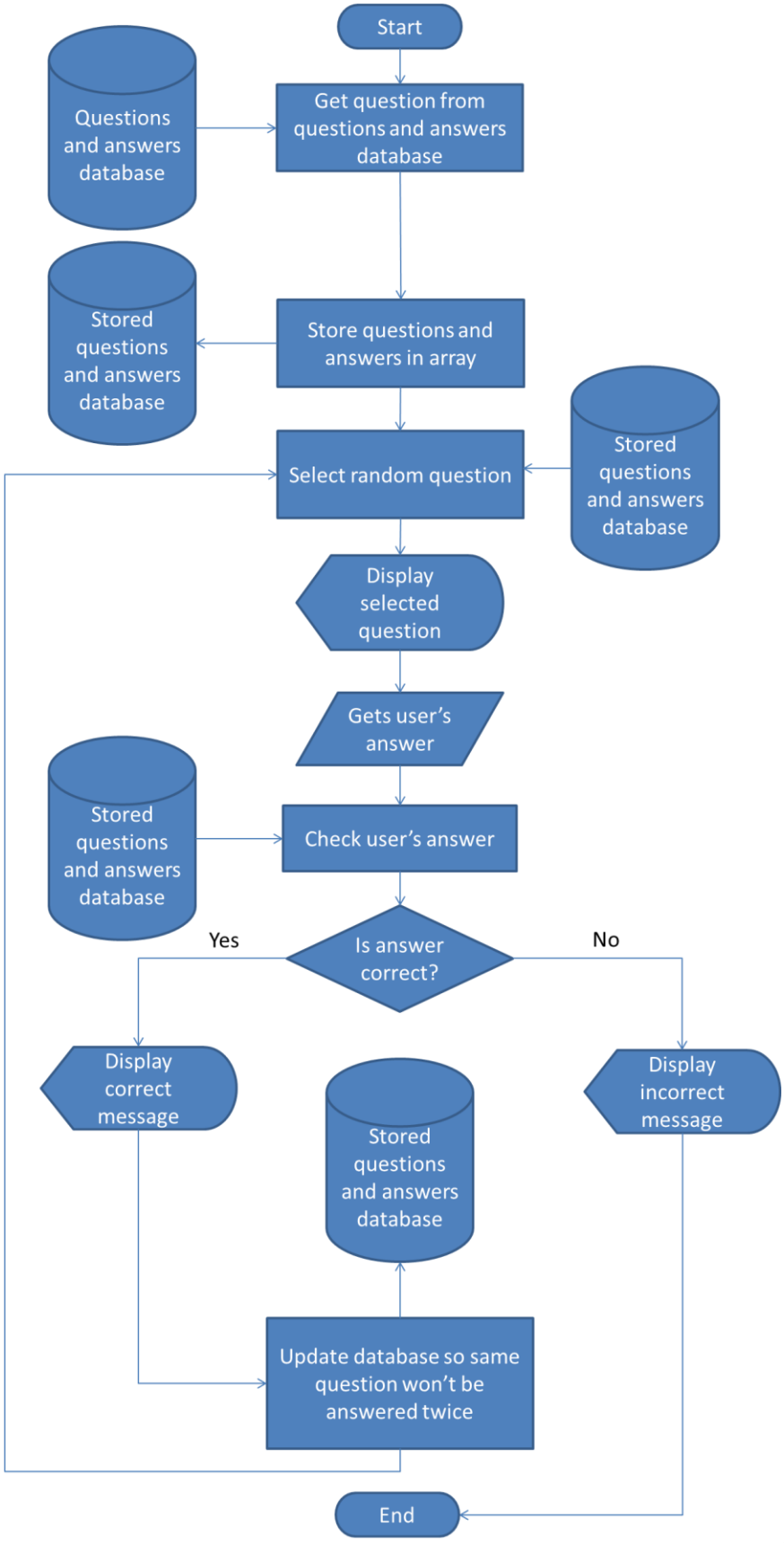
2.1 System overview

The program I will be creating is for a Physics teacher to help students visualize the motion of the ball object. It is meant to be an accurate representation of how a ball would normally behave as it passes through the fluid of choice. The program would be able to calculate many properties of the system such as the up-thrust acting on it, the viscous drag and the ball's speed etc. It will also offer a quiz mode where the system will provide a few questions for the user to answer and check if they got it correct.

2.2 System flowchart



System flowchart for simulator



System flowchart for quiz

2.3 Overall system design

Inputs

- Properties of ball
  - Radius
  - Density
  - Material
- Properties of fluid
  - Fluid
- Simulator inputs
  - Be able to run the animation after a fluid is chosen
  - Be able to pause the animation at any time
  - Be able to click a button to export data
  - Click a button to access a different mode
  - Click the main menu hyperlink to go back to the main menu
  - Be able to click the answer button to check their answers

Process

The processes below are for each timer tick (or for each frame)

- Calculating the current velocity of the ball
- Calculating the acceleration of the ball
- Calculating the resultant force acting on the ball
- Calculating the viscous drag acting on the ball

These processes happen only once after changing the properties of the ball (or fluid in the case of up-thrust)

- Calculate weight
- Calculate mass
- Calculate volume
- Calculate up-thrust

Storage

- Program reads in a database of questions and questions which the program stores in an array and which the program can randomly choose from
- Program may be able to export the data from the simulation

Outputs

- Properties of ball
  - Viscous drag
  - Current velocity
  - Terminal velocity
  - Up-thrust

  - Weight
  - Volume
  - Fluid density
  - Fluid viscosity
  - Time since ball has entered fluid
  - Table containing simulation details
  - Image of the ball
- In quiz
  - The question randomly selected from array
  - Sound effects
  - The score
  - An output displaying the correct answer if the user got it wrong
- In main menu
  - Sound effects

2.4 Modular structure

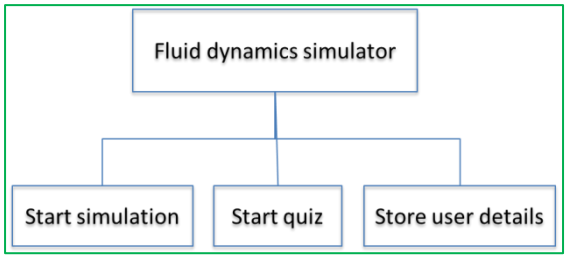
0 Fluid dynamics simulator

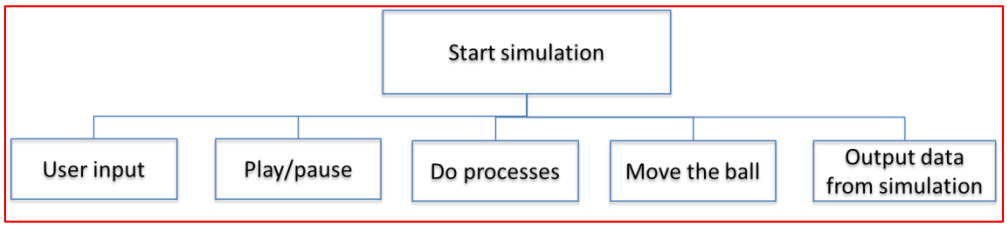
1 Start simulation

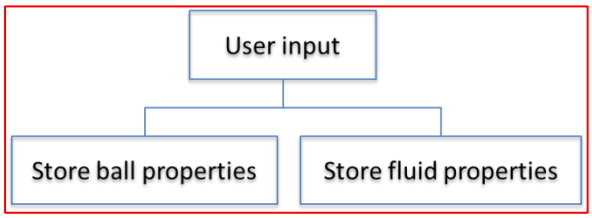
1.1 User inputs required data
1.1.1 User inputs radius of ball
1.1.2 User inputs density of ball
1.1.3 User selects fluid
1.1.3.1 Value is validated i.e. check whether it is blank as there is no default fluid
1.2 User presses run/pause button
1.3 Output simulation variables
1.3.1 Simulation calculates variables
1.3.1.1 Simulation calculates viscous drag
1.3.1.2 Simulation calculates current velocity
1.3.1.3 Simulation calculates terminal velocity
1.3.1.4 Simulation calculates up-thrust
1.3.1.5 Simulation calculates mass
1.3.1.6 Simulation calculates weight
1.3.1.7 Simulation calculates volume
1.3.1.8 Simulation calculates resultant force
1.3.1.9 Simulation calculates acceleration
1.3.1.10 Simulation updates time
1.4 Animate ball through fluid
1.4.1 Get new location
1.4.2 Move ball to new location
1.5 Output data if user wishes to

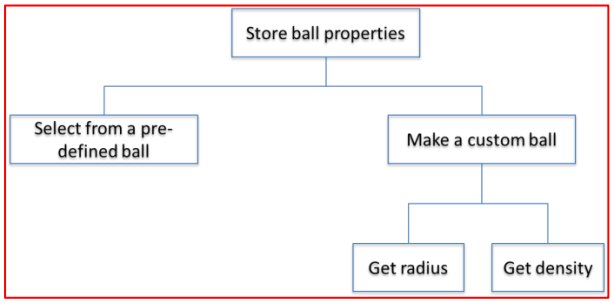
2 Start quiz

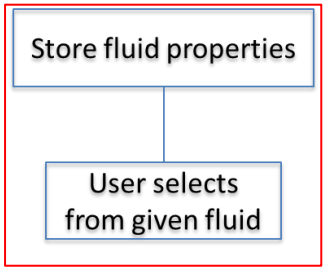
2.1 Loads database of questions and answers from file
2.2 Copies them to array in program
2.3 Display a random question on screen
    2.3.1 Choose a random number
2.4 Checks user input with answer
2.5 Produce correct message when user gets it correct
2.6 Produce negative message when user gets it incorrect and outputs correct answer

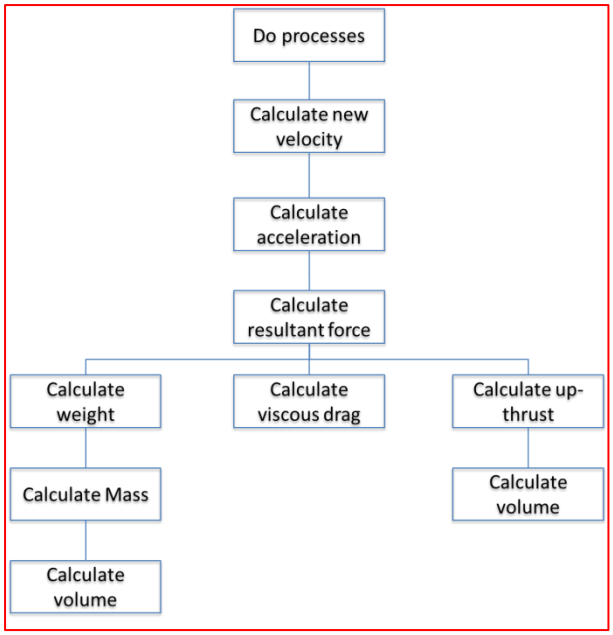


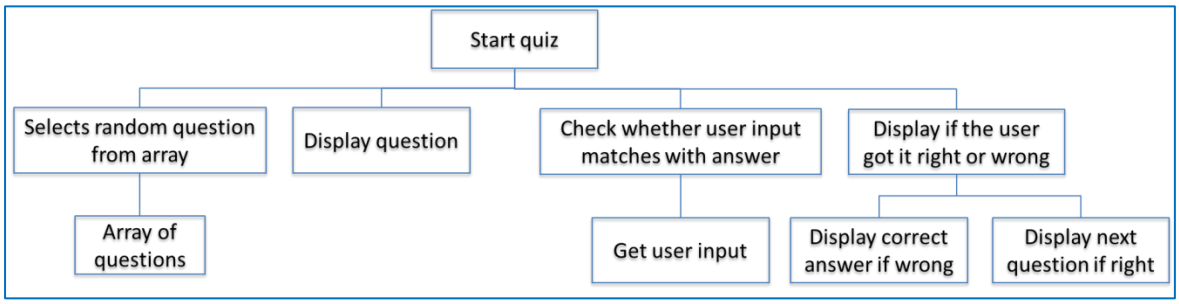












2.5 Record of database structure

Name	Description	Data type	Range/size	Example	Validation check	Validation details
BallDensity	The density which is given to the ball	Single	2-10 gcm^{-3}	6gcm^{-3}	Range check	Uses slide bar to limit the range of inputs allowed. Also does not allow erroneous data to be entered like other data types which can be entered in a textbox
Time	Shows the time passed since the	Single	0-0.5 seconds	0.2 seconds		

	ball was dropped or when it hit the fluid					
Fluid	Stores the fluid currently being used	String	Glycerol, Sunflower oil, Coconut oil, Olive oil	Glycerol	Drop down menu	Limits the user's inputs by giving them a set number of choices the user can choose from which are all valid
Material	Stores the material of the ball currently being used	String	Steel, glass	Steel	Drop down menu	Limits the user's inputs by giving them a set number of choices to choose from which are all valid
Radius	Stores the radius of the ball	Integer	2mm-8mm	6mm	Range check	Limits the range of inputs allowed. This is to make sure the size of the ball isn't too big so it won't fit inside the tube and that no erroneous data is entered e.g. a string
Distance	Determines how much the ball should move down per 0.001 seconds	Double	0-0.0211m	0.004m		
Current Velocity	Shows the	Double	0-21.1ms^{-1}	0.04ms-^{1}		

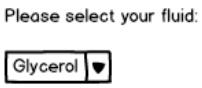
	distance the ball should move by in a given time. The rate of change of displacement.					
Terminal Velocity	Shows the constant velocity which the ball is going at.	Single	0.01 – 0.5ms^{-1}	0.3ms^{-1}		
SteelDensity	A constant for the density of steel	Single	Fixed	8.05gcm^{-3}		
MarbleDensity	A constant for the density of marble	Single	Fixed	2.40 gcm^{-3}		
GlycerolViscosity	A constant for the coefficient of viscosity of glycerol	Single	Fixed	1.412 Pas		
SunFlowerOilViscosity	A constant for the viscosity of sunflower oil	Single	Fixed	0.4914 Pas		
CoconutOilViscosity	A constant for the viscosity of coconut oil	Single	Fixed	0.06 Pas		
OliveOilViscosity	A constant for the viscosity of Olive oil	Single	Fixed	0.1075 Pas		
GlycerolDensity	A constant for the density of glycerol	Single	Fixed	1.26g/cm^{3}		
SunFlowerOilDensity	A constant for the density of sunflower	Single	Fixed	0.93g/cm^{3}		

	oil					
CoconutOilDensity	A constant for the density of coconut oil	Single	Fixed	0.924g/cm^3		
OliveOilDensity	A constant for the density of Olive oil	Single	Fixed	0.86g/cm^3		

2.6 Validation

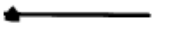
Fluid:

I will use a drop down menu for the type of fluid which the user wants. This is because the data is a string so sliders would not be logical to use and there is a list of pre-set fluids which the client wants to use. These fluids are the ones most commonly found in doing actual experiments. An example of how the drop down menu would look like is:



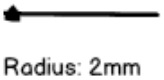
Density:

I will use a slider object for the density. This means I can control the range by limiting what the user inputs so that they don't put in a density that would cause the object to float or a density too heavy it would just sink very quickly. It also means that the user can't type in erroneous data like a string for the density of the ball. An example of how the density slider would look like is shown below:



Radius:

I will use a slider object to allow the user to change the radius. Not only is this aesthetically pleasing but it would allow the user to see the radius of the ball change as he moves the slider. This will also prevent the user from entering erroneous data like strings or a radius that is too small or large which might make the size of the ball go off the screen. An example of how the radius slider would look like is shown below:



(All the pictures of validation methods are taken from the user interface design section on page 53):

2.7 Hardware specification

Hardware	Description
Mouse	This enables the user to move the cursor on the screen around. The user can also left click on it to select buttons or click and hold to drag objects like sliders.
Keyboard	This allows the user to input their answers to the quiz.
Visual Display Unit (VDU)	This hardware provides a graphical display to the user. The user would then be able to see the form and see the visual animation. The user can also see the outputs such as the velocity of the object etc. from here.
Hard Disk Drive (HDD)	This is required to store the program and all the data needed.
CD – Drive	This enables the user to install the program from the CD.
USB stick	This is needed in case the user does not have a CD – drive on his/her laptop. The user can put the CD into a school computer which all have CD –drives, then copy and paste the files onto the USB stick so they can still install the program onto their laptops via this method instead.

2.8 File organisation & processing;

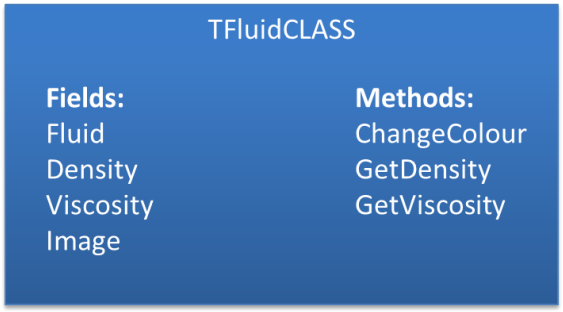
The files I need will be stored as CSV files. There will be 3 main files:

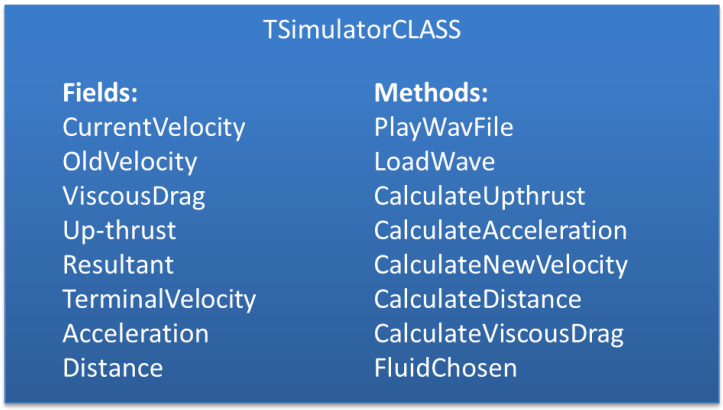
1. Database containing all the questions and answers for the quick quiz
2. Database containing all the questions and answers for the calculation quiz
3. An output of the results of the simulation as a table

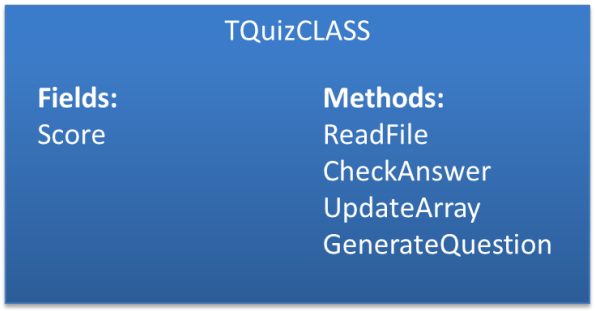
For file no.1 and no.2 the method of access needs to be random. This is because the user needs to be given a random question each time or else they can remember the answer to what question is coming up next if they got it wrong the first time. The file is to provide students with some potential exam questions and test their knowledge. Each time the user gets a question right, one record will be updated. Actually the space in the record will get deleted and then all the other records below it will be moved one position up. This means that the same question won't be presented twice to the user. Examples of the finished question files can be seen in the appendix (page 123).

For file no.3 serial access file organisation would be appropriate. This is because the records in the table would need to be in order so that they can see the values of the variables in a chronological order as the simulation progressed. The file is to allow the students to analyse data afterwards. An example of how the finished file looks like can be found in the appendix (page 123 and 124)

2.9 Object diagrams

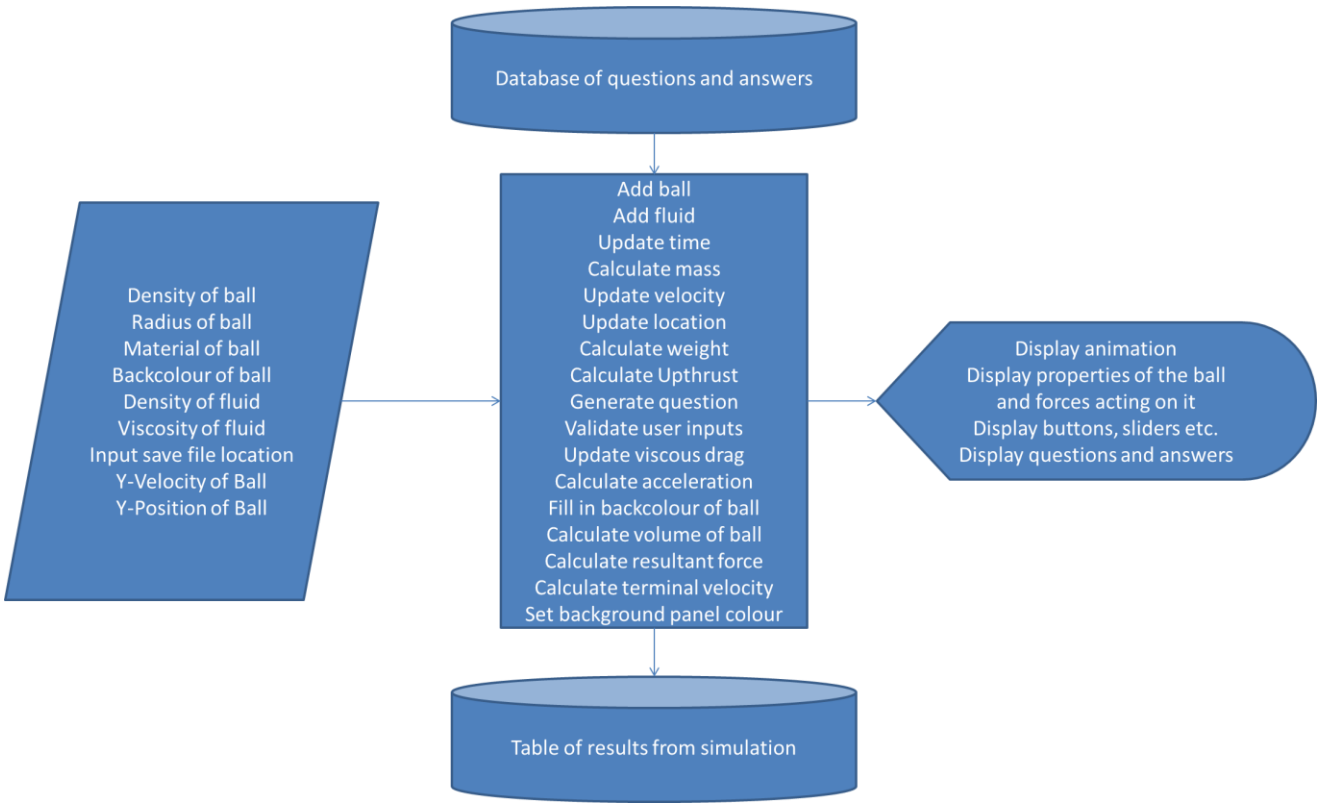






2.10 Database design and E-R model

IOPS



2.11 Storage media & format

Size of program

My program won't take up that much space since it's a simulation program with basic graphics instead of a database project. I think it should only need to take up 2MB in a storage device's memory. I am planning to distribute my program through a CD-ROM.

Size of simulation data files

The simulation data files would be stored as CSV files which don't take up that much memory. The CSV files shouldn't exceed more than 50KB. The CSV files would be written onto the user's hard disk drive or solid state drive once the user decides to export the data from the simulation.

2.12 Algorithms using pseudocode

Algorithm to calculate Volume

BEGIN

'radius in units of mm

Volume = (4*pi*(radius/1000) ^3)/3

END

Algorithm to calculate Mass

BEGIN

'Have to multiply the density by 1000 to get it from gcm^{-3} to kgm^{-3}

Mass = volume * density * 1000

END

Algorithm to calculate Weight

BEGIN

'Calculates weight using mass

Weight = Mass*g

END

Algorithm to calculate viscous drag

BEGIN

'Calculates viscous drag acting on ball using Stokes' law equation

ViscousDrag = 6*pi*v*r* η

END

Algorithm to calculate up-thrust

BEGIN

'Denisty of fluid in gcm^{-3} so have to convert it to kgm^{-3}

Upthrust = volume * DensityOfFluid * 1000 * g

END

Algorithm to calculate new velocity of ball through fluid

BEGIN

'Calculate resultant force using free body force diagram

Resultant = Weight – (ViscousDrag + Upthrust)

'Gets acceleration from Newton's second law

Acceleration = Resultant/Mass

'Gets new velocity using SUVAT

v = u + at

'Updates with new velocity

u = v

END

Algorithm to calculate the distance moved by the ball

BEGIN

'Gets distance by doing the instantaneous velocity multiplied by the time interval for each update

Distance = currentVelocity * time

BallPosition = BallPosition + Distance

END

Algorithm to calculate terminal velocity of ball through fluid

BEGIN

$(2*radius^2*g*(densityOfSolid – densityOfFluid))/(9*viscosity)$

END

Algorithm to check the whether the ball is going out of tube

BEGIN

REPEAT

'Checks if the ball's position + the height/radius of the ball to account for the different radii of the ball exceeds the position of the bottom of the tube

IF BallPosition + BallHeight > 438 THEN

```
                Timer = False
            ENDIF
        UNTIL BallPosition + Ball Height >438
END
```

Algorithm to export data

```
BEGIN
        INPUT FileLocation
        IF FileLocation <> "" THEN
            'Outputs the labels
            OUTPUT "Time (s), Velocity (m/s), Resultant force (N), Viscous drag (N), Acceleration
            (m/s^2), Terminal velocity, m/s, Upthrust, N, Radius, mm, Weight, N, Volume, m^3,
            Density of ball, g/cm^3, Fluid, Fluid density, g/cm^3, Viscosity, Pas"
            'Outputs the pieces of data for the corresponding labels
            For x = 1 to Time*1000 + 1
                OUTPUT data(0, x)
                OUTPUT data(1, x)
                OUTPUT data(2, x)
                OUTPUT data(3, x)
                OUTPUT data(4, x)
                OUTPUT new line
            NEXT
            'Informs the user whether the data exportation was successful or not
            OUTPUT "data exported!"
        ELSE
            OUTPUT "invalid file location"
        ENDIF
END
```

Algorithm for reading files for quiz

```
BEGIN
    'Tells the program which file to read in depending on which type of quiz the user selected
    INPUT quizType
    IF quizType = "Calculate" THEN
        READ CalculationQuizFile
    ELSE
        READ QuickQuizFile
    ENDIF
    'Reads in each question from the file. A line being one question.
    REPEAT
        Questions(x) = ReadLine
        x = x+1
    UNTIL EndOfFileReached
END
```

Algorithm for checking answers for quiz

```
BEGIN
    'Keeps checking whether the user inputted the correct answer or not
    REPEAT
        INPUT answer
        IF answer = storedAnswer THEN
            OUTPUT positiveSound
            score = score + 10
        ENDIF
    UNTIL answer <> storedAnswer OR questionsAnswered = noOfQuestions
    IF answer <> storedAnswer
        'Once the user gets the question wrong, the system will output a negative sound, inform the user of the correct answer and their final score
        OUTPUT negativeSound
        OUTPUT storedAnswer
        OUTPUT score
    ELSE
        'If the user manages to do all the questions correctly, then a congratulations message will appear then output their score which will also be the maximum possible score
        OUTPUT "Well done you finished the game!"
        OUTPUT score
END
```

Algorithm for outputting random question for quiz

```
BEGIN
    'Sets the initial position to look for the comma indicating when the question starts
    firstCommaPos = 1
    'Get a question number between 1 and the number of questions
    QuestionNo = Rand*noOfQuestions + 1
    'Finds the position of the first comma indicating the start of a question
    REPEAT
        IF mid(Questions(QuestionNo), firstCommaPos, 1) = "," THEN
            firstPos = true
        ELSE
            firstCommaPos = firstCommaPos + 1
        ENDIF
    UNTIL firstPos = true
    'Finds the position of the second comma indicating the start of the answer and end of the question
    secondCommaPos = firstCommaPos + 2
    REPEAT
        IF mid(Questions(QuestionNo), secondCommaPos, 1) = "," THEN
            secondPos = true
        ELSE
            secondPos = secondPos + 1
        ENDIF
    UNTIL secondPos = true
    OUTPUT mid(Questions(questionNo), firstCommaPos + 2, secondCommaPos - (firstCommaPos +2))
END
```

(Please see key terminology page for meanings of symbols and Physics terms)

2.13 User interface (I/O)

Types of user interface

GUI

Advantages	Disadvantages
It can create forms easily	Requires more memory resource.
It is easier for the user to visualize this	There are many buttons and it may take some time to navigate around
It is suitable for animations	

Menu driven

Advantages	Disadvantages
Easy to use as one menu leads nicely to another which is related to it	If there are too many menus for the user to navigate through, the user might get lost

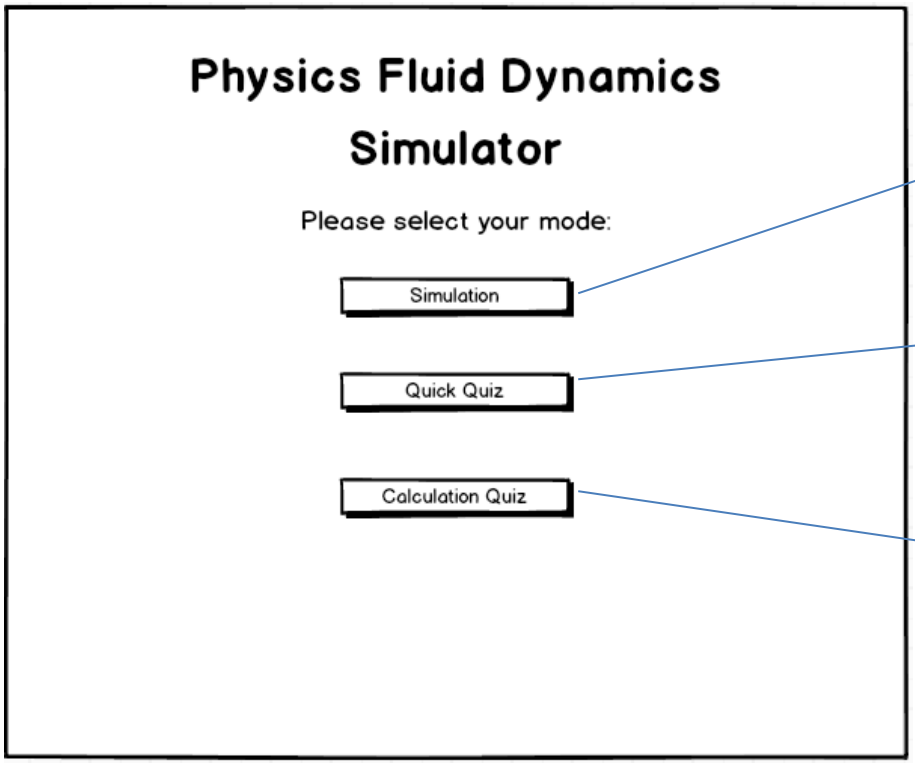
CLI

Advantages	Disadvantages
It can be powerful with experienced user and programmer	It might be hard remembering all the syntax to do the programming
	There is no media making the idea of a graphical simulator hard to create

My choice

The type of user interface I decided to use was GUI. This is because I felt that that would be the most appropriate type of user interface for an animation program like mine.

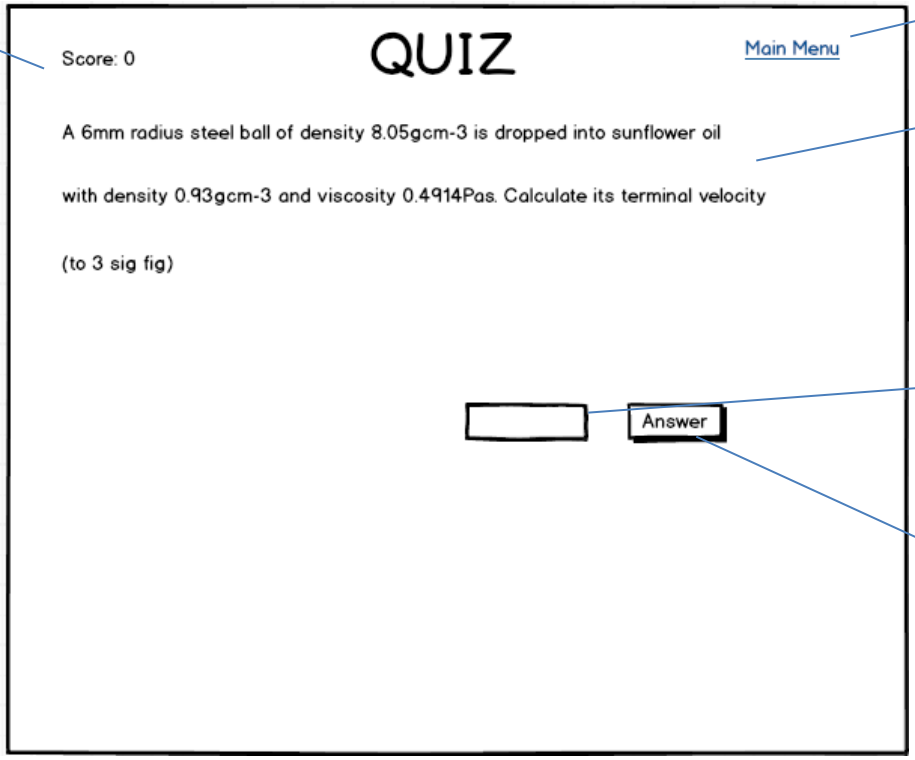
The following pictures are some designs of my user interface:



Simulation button – brings the user to the simulation form

Quiz quiz button – brings the user to the quick quiz form

Calculation quiz button – brings the user to the calculation quiz form



Main menu link – brings the user back to the main menu form

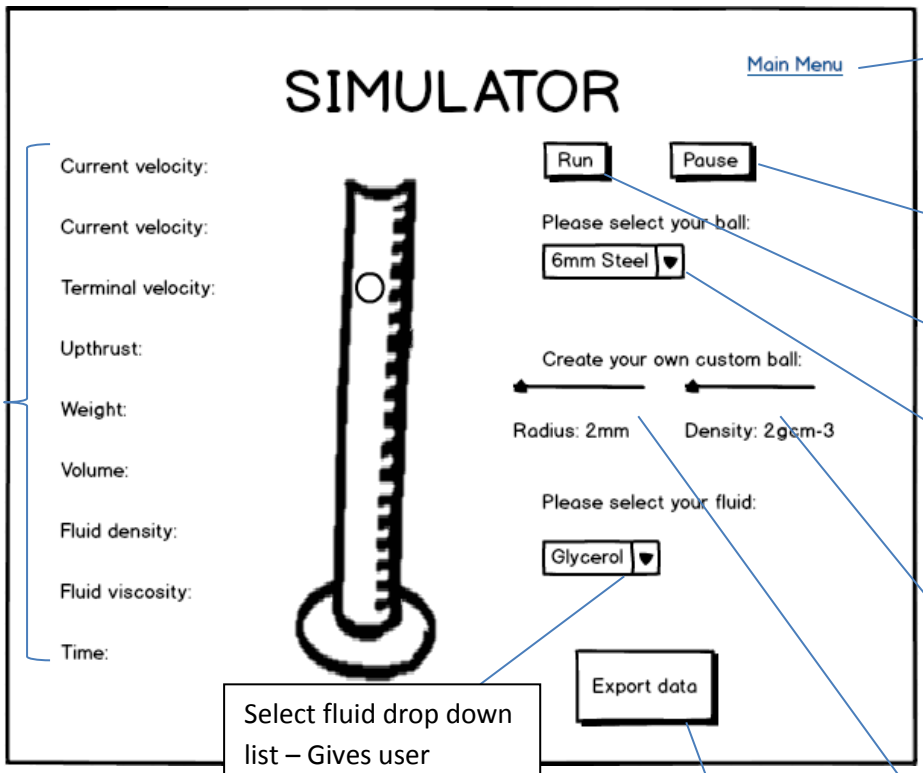
Score – shows the user's current score

Question – Displays the question in a label which is presented to the user

Answer textbox – Allows the user to input their answer here

Answer button – Checks the answer the user inputted in the answer textbox

(User interfaces the same for both quiz types. Only the type of questions is different)



Main menu link – brings the user back to the main menu form

Pause button – Pauses the simulation when clicked

Run button – Runs the simulation when clicked

Labels – showing the values of different properties in the simulation

Select ball drop down list – Gives the user a set of pre-defined balls to choose from

Density slider – changes the density of the ball

Select fluid drop down list – Gives user options to select a pre-defined fluid from

Export data button – Allows the user to export the simulation's data

Radius slider – changes the radius of the ball

Examples of some screenshots in how the user interface would look like
Pictures are generated from balsamiq (https://balsamiq.com)

2.14 Security and integrity of data

My program is a simulator program and not a database program; therefore the data protection act 1998 does not really apply to my program as I won't need to store any personal information about the users. I am happy for my client to distribute the software freely to the school's Physics students or anyone else she may like so copyright isn't a big issue for me. However, I do not want other people to modify the program so the application is given to them without the source code.

The integrity of my data is strong since there are a limited number of places where the user can type in anything they want in a textbox. In the main menu, there are just buttons which the user can click, which all of them works. In the simulator mode, everything is validated by either sliders or things like drop down menus. If the user wants to run or pause the simulation, they simply click on the buttons. If they click on the pause button while it is paused nothing happens. If they want to change the size or density of the ball they can use the drop down menu or sliders which have a fixed range of inputs, where all of them are valid. The selection of a fluid is also selected by a drop down menu. If the user wishes to export the data, they can type in anything they want and the program won't crash. If the user does not type in anything then the input box just closes after the user clicks ok or if there was

something inputted by the user then a message saying the data was exported or an error message will come up.

In the quiz mode, the user can type in anything they want in the answer textbox and the system would only recognise it as right or wrong. The answers the user types in is also not case sensitive because my program converts all letters to upper case and compares it with the answer also written in upper case, therefore both “Laminar flow” and “LaMINar FloW” would both be correct. In conclusion, I can say that the integrity of my data is sufficient for my program.

2.15 System security

The pictures will be stored in the exe file so the users don’t have access to them. This would prevent them from changing any of the files or accidentally deleting them. I have let the users have access to the questions and answers for the quiz in case the teacher wants to add a question, change the answer to a question or even delete a question. This allows the teacher to have the flexibility to alter the quiz as time goes on because the syllabus might change or they find some questions more useful than others. A backup of these files would be stored in another location e.g. somewhere else on the teacher’s hard disk drive or in the cloud in case the files are accidentally changed or deleted. I have thought of encrypting the files with password protection to only allow the teachers and not the students to access and change the files for the quiz. However, this would mean that my program wouldn’t be able to read in the file straight away so I have decided to leave it unprotected. I will discuss this further in my appraisal under possible extensions.

2.16 Test strategy

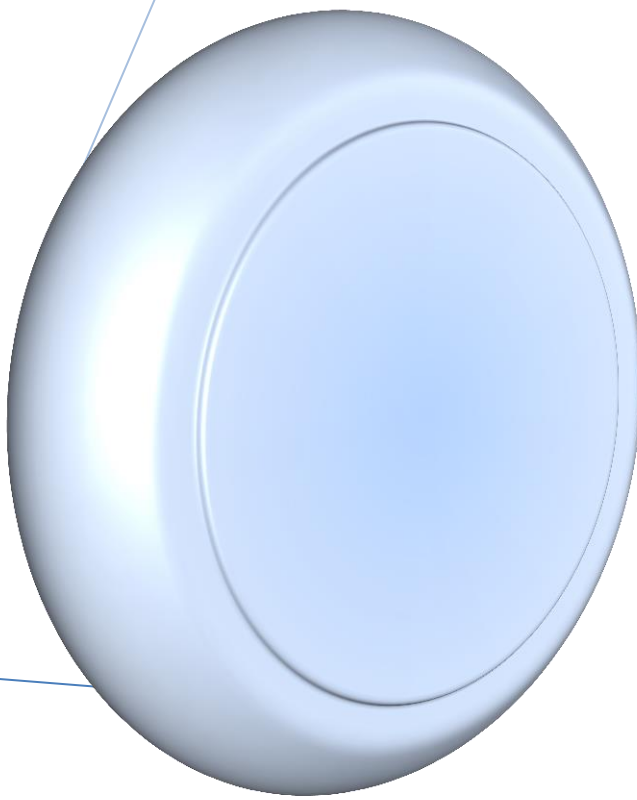
Black box testing – Black box testing is a method of software testing that examines the functionality of an application (e.g. what the software does) without peering into its internal structures or workings. I will do the black box testing by entering some inputs and see if the correct output is generated. I will run the program to see if the objective to be met can be seen or not and if it can, the objective has been achieved

White box testing – White box testing is a method of testing software that tests internal structures or workings of an application. I will do white box testing with each of the modules to check whether it is calculating the correct values. I will need to do the calculations by hand to see whether the output matches my answer. I will also have to create trace tables to check whether my logic is correct.

Beta testing – Beta testing

Objectives	Test plan
It will make an animation of the ball in fluid	The user will be able to observe the ball moving positions in the fluid (test no.7 of the simulation mode in testing)
The ball will stop once it hits the bottom of the container	The ball won't go through the tube in a simulation (test no.10 of the simulation mode in testing)
The user can change the fluid therefore changing the viscosity and density of the fluid	The values of the density and viscosity of the fluid outputted to the user should change as the user selects a different fluid (test no.4 of the simulation mode in testing)
The user can change the ball i.e radius, density etc.	The size of the ball should update in real time as the user changes the radius (test no.2 and 5 in the simulation mode in testing)
The user can play a quiz mode	It should bring the user to the appropriate screen when clicked (test no.1 of the main menu mode in testing)
The user will be able to know the current speed	The current speed should be displayed (test no.13 of the simulation mode in testing)
The user will be able to know the terminal velocity	The terminal velocity should be displayed and can be checked by hand calculation (test no.13 of the simulation mode in testing)
The user will be able to know the density of the fluid	The density should be displayed. The density can be checked with the data dictionary to see whether the fluid and density corresponds (test no.4 of the simulation mode in testing)
The user will be able to know the volume of the ball	The program would display the volume. The volume could be checked by hand after knowing the radius (test no.2 of the simulation mode in testing)
The user will be able to know the up-thrust acting on the ball	The up-thrust value would be displayed. It can be checked by doing the calculation by hand (test no.13 of the simulation mode in testing)
The colour of the fluid would change as a new fluid is selected	When the user selects a fluid, the colour of the fluid in the tube will change to represent that fluid (test no.4 of the simulation mode of testing)
The size of the ball would change as the radius property is changed	As the user scrolls the radius slider bar or selects a new ball, the size of the ball on screen will change (test no.5 of the simulation mode in testing)
The picture of the ball would change if the material of the ball is changed (i.e. glass, steel or generic material which represents the density of the density slider bar)	When the user selects a ball of different material from the drop down list, its picture will change. If the ball is currently a steel or marble ball and the user changes the value of the density slider, then the image should change as well (test no.6 of the simulation mode in testing)
A positive result will be returned when the user gives the correct answer in play mode	Answer the question correctly and see the appropriate message displayed

	(test no.1 of the quiz mode in testing)
A negative result will be returned when the user gives a wrong answer in play mode	Answer the question incorrectly and see the appropriate message displayed (test no.2 of the quiz mode in testing)
Sound effects will be heard at the appropriate time	When a button is clicked, sound effects will be heard (test no.2 of the main menu mode and test no. 4 of the quiz mode in testing)
The time elapsed will be displayed	Watch the time go up (test no.13 of the simulation mode in testing)
The user will be able to choose a fluid from glycerine, sunflower oil, coconut oil and olive oil	After clicking on the drop down menu, see the different types of fluid displayed (test no.4 of the simulation mode in testing)
The user will be able to choose standard balls which are 8mm steel ball, 6mm steel ball, 8mm marble ball, 6mm marble ball	After clicking on the drop down menu, see the different types of balls displayed (test no.6 of the simulation mode in testing)
The animation of the ball would be smooth	When the simulation is running, there shouldn't be lag (test no.7 of the simulation mode in testing)
The results will be outputted to 2 - 5 significant figures on the simulation	The results will be outputted to 2 - 5 significant figures (test no.13 of the simulation mode in testing)
The data of the simulation would be outputted into a table and stored as an external file	You can access and view the file of results after simulation has ended (test no.13 of the simulation mode in testing)
The user can return to the main menu to switch between different modes	The user is able to click on the main menu button and return to the main menu to select a different mode (test no.9 of the simulation mode in testing)
The system will keep track of the user's score based on how many he/she got right in quiz mode	The user will be able to see the score in the top left hand corner of the quiz mode (test no.3 of the quiz mode in testing)



Technical solution

Fluid dynamics simulator

In this section of the coursework, I aim to justify my project's complexity.

Victor Sim
3/26/2015

3. Technical solution

3.1 Complexity

The complexity of my problem mainly lies in the mathematical problems in how to calculate all the values to simulate this accurately. This would require A-level physics and maths knowledge (FC6 – Physics modelling). First of all I will have to make use of a free body force diagram to work out all the forces acting on the ball. The force include up-thrust, weight and viscous drag. I can calculate viscous drag using Stokes' law ($F=6\pi r\eta v$) and I can calculate the up-thrust using some density, volume equations. After this I can find out my resultant force. I can work out the mass of the ball by making use of its volume which is $4\pi r^3/3$ and its density. Using Newton's second law, $F=ma$ and the mass and resultant force which I have calculated in the previous part, I can find the acceleration of the ball. I would have to assume constant acceleration for a short time period in order to use SUVAT equations. This is still a good estimate as the time interval is only 0.001 seconds.

My problem is also complex because my simulation is time based (FC4 – Time based simulation). The time displayed in the simulation would be how the ball would behave like at that time in the real world. For example, if it says 0.025 seconds in the simulation then the ball in a fluid in a real experiment after 0.025 seconds would also have similar values to those produced in the simulation e.g. speeds in real world and simulation will be similar.

The problem is also complex because it would require object orientated programming (FC2 – User defined objects/classes) featuring data encapsulation. I will need to create a ball object, a fluid object, a quiz object and a simulator object using the classes I will make myself. The data encapsulation will prevent other procedures using the data incorrectly since all the data and source code that manipulates data is contained inside a self-contained entity. The user defined classes I have made are:

```
Public Class Ball
    Private p_radius As Integer
    Private p_density As Double
    Private p_top As Double
    Private p_material As String
    Private p_volume As Double
    Private p_mass As Double
    Private p_weight As Double
    Private p_image As PictureBox
    Const g = 9.81
    Const steelDensity = 8.05
    Const marbleDensity = 2.4

    Public Sub ChangeColour(fluid)
        'Changes the backcolour of the ball according to the fluid so that the
backcolour of the ball matched the fluid colour
        Select Case fluid
            Case "Glycerol"
                p_image.BackColor = Color.FromArgb(255, 215, 215, 215)
            Case "Sunflower oil"
                p_image.BackColor = Color.Yellow
            Case "Coconut oil"
                p_image.BackColor = Color.FromArgb(255, 225, 230, 30)
            Case "Olive oil"
                p_image.BackColor = Color.FromArgb(255, 200, 200, 60)
```

```
            Case Else
                p_image.BackColor = Control.DefaultBackColor
        End Select
    End Sub

    Public Sub CalculateVolume(radius As Integer)
        'Radius is in units of mm
        p_volume = (4 * System.Math.PI * (radius / 1000) ^ 3) / 3
    End Sub

    Public Sub Calculatemass(volume, density)
        'density is in g/cm^3 so have to convert it to kg/m^3 by multiplying by 1000
        p_mass = volume * density * 1000
    End Sub

    Public Function Getweight()
        'Calculates the weight using mass
        p_weight = p_mass * g
        Return p_weight
    End Function

    Public Function GetDensity()
        Dim density As Double
        'Assigns the density according to what material and returns the density
        If p_material = "steel" Then
            density = steelDensity
        ElseIf p_material = "marble" Then
            density = marbleDensity
        End If
        Return density
    End Function

    Property image As PictureBox
        Get
            Return p_image
        End Get
        Set(value As PictureBox)
            p_image = value
        End Set
    End Property

    Property Material As String
        Get
            Return p_material
        End Get
        Set(value As String)
            p_material = value
        End Set
    End Property

    Property Y As Double
        Get
            Return p_top
        End Get
        Set(value As Double)
            p_top = value
        End Set
    End Property

    Property radius As Integer
        Get
```

```
            Return p_radius
        End Get
        Set(value As Integer)
            p_radius = value
        End Set
    End Property

    Property density As Double
        Get
            Return p_density
        End Get
        Set(value As Double)
            p_density = value
        End Set
    End Property

    Property volume As Double
        Get
            Return p_volume
        End Get
        Set(value As Double)
            p_volume = value
        End Set
    End Property

    Property weight
        Get
            Return p_weight
        End Get
        Set(value)
            p_weight = value
        End Set
    End Property

    Property mass
        Get
            Return p_mass
        End Get
        Set(value)
            p_mass = value
        End Set
    End Property
End Class
```

```
Public Class Fluid
    Private p_fluid As String
    Private p_density As Double
    Private p_viscosity As Double
    Private p_image As PictureBox
    Const GlycerolDensity = 1.26
    Const SunFlowerOilDensity = 0.93
    Const CoconutOilDensity = 0.924
    Const OliveOilDensity = 0.86
    Const GlycerolViscosity = 1.412
    Const SunFlowerOilViscosity = 0.4914
    Const CoconutOilViscosity = 0.06
    Const OliveOilViscosity = 0.1075

    Public Sub Changecolour()
        'Loads a tube with a different colour depending on what fluid is selected
```

```
        Select Case p_fluid
            Case "Glycerol"
                p_image.Image = My.Resources.Glycerol_test_tube
            Case "Sunflower oil"
                p_image.Image = My.Resources.Sunflower_test_tube
            Case "Coconut oil"
                p_image.Image = My.Resources.Coconut_oil_test_tube
            Case "Olive oil"
                p_image.Image = My.Resources.Olive_oil_test_tube
            Case Else
                p_image.Image = My.Resources.Test_tube
        End Select
    End Sub

    Public Function GetDensity()
        'Gets the density of the fluid and returns it according to what is the
selected fluid
        Select Case p_fluid
            Case "Glycerol"
                p_density = GlycerolDensity
            Case "Sunflower oil"
                p_density = SunFlowerOilDensity
            Case "Coconut oil"
                p_density = CoconutOilDensity
            Case "Olive oil"
                p_density = OliveOilDensity
        End Select
        Return p_density
    End Function

    Public Function GetViscosity()
        'Gets viscosity of the fluid and returns it according to what is the selected
fluid
        Select Case p_fluid
            Case "Glycerol"
                p_viscosity = GlycerolViscosity
            Case "Sunflower oil"
                p_viscosity = SunFlowerOilViscosity
            Case "Coconut oil"
                p_viscosity = CoconutOilViscosity
            Case "Olive oil"
                p_viscosity = OliveOilViscosity
        End Select
        Return p_viscosity
    End Function

    Property image As PictureBox
        Get
            Return p_image
        End Get
        Set(value As PictureBox)
            p_image = value
        End Set
    End Property

    Property fluid As String
        Get
            Return p_fluid
        End Get
        Set(value As String)
            p_fluid = value
```

```
        End Set
    End Property

    Property density As Single
        Get
            Return p_density
        End Get
        Set(value As Single)
            p_density = value
        End Set
    End Property

    Property viscosity As Single
        Get
            Return p_viscosity
        End Get
        Set(value As Single)
            p_viscosity = value
        End Set
    End Property
End Class
```

```
Public Class Simulator
    Private p_OldVelocity As Double = 0
    Private p_CurrentVelocity As Double
    Private p_TerminalVelocity As Double
    Private p_ViscousDrag As Double
    Private p_Upthrust As Double
    Private p_Resultant As Double
    Private p_Acceleration As Double
    Private p_Distance As Double
    Const g = 9.81
    Const time = 0.001

    Public Sub CalculateUpthrust(volume, densityOfFluid)
        'Caclulates up-thrust using volume and density
        'Multiplied by 1000 to get it in SI units
        p_Upthrust = volume * densityOfFluid * 1000 * g
    End Sub

    Public Sub CalculateAcceleration(Weight, Mass)
        'Calculates the resultant force
        p_Resultant = Weight - (p_ViscousDrag + p_Upthrust)
        'Uses Newton's 2nd law to get the acceleration
        p_Acceleration = p_Resultant / Mass
    End Sub

    Public Sub CalculateNewVelocity()
        'Uses the SUVAT equation v = u + at
        p_CurrentVelocity = p_OldVelocity + p_Acceleration * time
        'Updates the old velocity to become the new velocity
        p_OldVelocity = p_CurrentVelocity
    End Sub

    Public Sub CalculateDistance()
        'Calculates the distance it should move by using distance = speed * time
        'Multiplies the time by 100000 to get noticable movement because time in order
of magnitude 10^-3 and velocities are very small
        p_Distance = p_CurrentVelocity * time * 100000
    End Sub
```

```
    Public Sub CalculateTerminalVeloctiy(radius, solidDensity, fluidDensity,
viscosity)
        'Calculates the terminal velocity using equation (2r^2(ρs – ρf))/(9η)
        'Divides and multiplies by 1000 to get it into SI units
        p_TerminalVelocity = (2 * (radius / 1000) ^ 2 * g * (solidDensity * 1000 -
fluidDensity * 1000)) / (9 * viscosity)
    End Sub

    Public Sub CalculateViscousDrag(radius, viscosity)
        'Calculates the viscous drag acting on the ball using Stokes' law
        p_ViscousDrag = 6 * System.Math.PI * p_CurrentVelocity * radius * viscosity
    End Sub

    Public Function FluidChosen(fluid) As Boolean
        'Checks if there is fluid chosen and returns true if there is or false if
there isn't
        FluidChosen = True
        If fluid = "" Then
            FluidChosen = False
        End If
        Return FluidChosen
    End Function

    Property currentVelocity
        Get
            Return p_CurrentVelocity
        End Get
        Set(value)
            p_CurrentVelocity = value
        End Set
    End Property

    Property oldVelocity
        Get
            Return p_OldVelocity
        End Get
        Set(value)
            p_OldVelocity = value
        End Set
    End Property

    Property viscousDrag
        Get
            Return p_ViscousDrag
        End Get
        Set(value)
            p_ViscousDrag = value
        End Set
    End Property

    Property Upthrust
        Get
            Return p_Upthrust
        End Get
        Set(value)
            p_Upthrust = value
        End Set
    End Property

    Property Resultant
```

```
        Get
            Return p_Resultant
        End Get
        Set(value)
            p_Resultant = value
        End Set
    End Property

    Property TerminalVelocity
        Get
            Return p_TerminalVelocity
        End Get
        Set(value)
            p_TerminalVelocity = value
        End Set
    End Property

    Property Acceleration
        Get
            Return p_Acceleration
        End Get
        Set(value)
            p_Acceleration = value
        End Set
    End Property

    Property Distance
        Get
            Return p_Distance
        End Get
        Set(value)
            p_Distance = value
        End Set
    End Property
End Class
```

```
Imports System.IO

Public Class Quiz
    Private p_score As Integer
    Private z As Integer
    Private x As Integer = 0
    Private Questions(50) As String
    Private questionNo As Integer
    Private audioDevice As New Microsoft.VisualBasic.Devices.Audio

    Public Sub ReadFile(quiztype)
        'Creates streamreader class
        Dim fileReader As StreamReader
        'Reads the short questions for QuickQuiz by default
        fileReader = New
StreamReader(My.Computer.FileSystem.SpecialDirectories.MyDocuments & "\Short
questions.csv")
        'If it is the CalculationQuiz thenr reads in the calculation questions
        If quiztype = "calculate" Then
            fileReader = New
StreamReader(My.Computer.FileSystem.SpecialDirectories.MyDocuments & "\Long
questions.csv")
        End If
```

```
        'Stores each line of the file in an array until end of file is reached
        Do
            Questions(x) = fileReader.ReadLine()
            x += 1
        Loop Until fileReader.EndOfStream
        'Closes filereader so it can read files again later
        fileReader.Close()
        'Redims the Questions array which was initially Questions(50) to how many
questions there actually are on the document to save memory space
        ReDim Preserve Questions(x)
    End Sub

    Public Sub CheckAnswer(ByVal Answer As String, ByRef Correct As Boolean)
        'Converts the use input to capital letters so that it is not case sensitive
        'Checks user's answer with answer part of the line
        If UCase(Answer) = Mid(Questions(questionNo), z + 2,
Len(Questions(questionNo)) - z + 2) Then
            'Executes these lines if the user got it right
            'Plays sound effect indicating the user got it right
            audioDevice.Play(My.Resources.Correct, AudioPlayMode.Background)
            'Updates score
            p_score += 10
        Else
            'Execute these lines if the user got it wrong
            'Plays a sound effect indicating the user got it wrong
            audioDevice.Play(My.Resources.Incorrect, AudioPlayMode.Background)
            Correct = False
            'Outputs the correct answer for user's knowledge
            MsgBox("Sorry you got that wrong! The correct answer was: " &
Mid(Questions(questionNo), z + 2, Len(Questions(questionNo)) - z + 2))
            'Outputs their score and a message depending on how high their score is
            If Score > 50 Then
                MsgBox("You got a score of: " & p_score & ". Well done!")
            ElseIf Score > 0 Then
                MsgBox("You got a score of: " & p_score & ". Good effort!")
            Else
                MsgBox("You got a score of: " & p_score & ". Need to work harder!")
            End If
        End If
    End Sub

    Public Sub UpdateArray()
        Dim y As Integer
        'Shifts the position of questions in the array below the selected question up
by a position of 1
        For y = questionNo To (x - 1)
            Questions(y) = Questions(y + 1)
        Next
        'Subtract 1 from the value of x to show how many questions are left remaining
        x -= 1
        'Redim preserves the array so that a 'blank' question would not be picked
        ReDim Preserve Questions(x)
    End Sub

    Function GenerateQuestion()
        Dim firstpos As Boolean = False
        Dim secondpos As Boolean = False
        Dim y As Integer
        'x = 0 when all questions answered
        If x = 0 Then
            MsgBox("Well done you finished the game!")
```

```
            MsgBox("You got a score of: " & p_score & ". Well done!")
            Return Nothing
        Else
            y = 1
            'Chooses a random question
            questionNo = CInt(Int(Rnd() * x))
            'Gets question part of the csv file
            Do
                If Mid(Questions(questionNo), y, 1) = "," Then
                    firstpos = True
                Else
                    y += 1
                End If
            Loop Until firstpos = True
            z = y + 2
            Do
                If Mid(Questions(questionNo), z, 1) = "," Then
                    secondpos = True
                Else
                    z += 1
                End If
            Loop Until secondpos = True
            'Outputs the question
            Return Mid(Questions(questionNo), y + 2, z - (y + 2))
        End If
    End Function

    Property Score As Integer
        Get
            Return p_score
        End Get
        Set(value As Integer)
            p_score = value
        End Set
    End Property
End Class
```

Some instances where I have declared objects of these classes are in the simulatorProgram where:

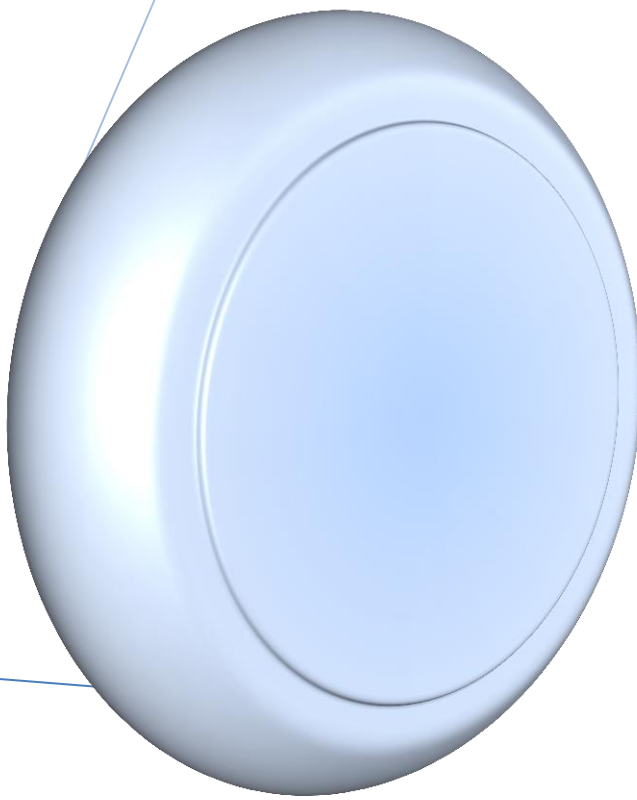
```
Public Class SimulatorProgram
    Dim myBall As New Ball
    Dim myFluid As New Fluid
    Dim mySimulator As New Simulator
```

And methods I have used from them, for example in the quiz program:

```
myQuiz.readFile("quick")
myQuiz.CheckAnswer(txtAnswer.Text, Correct)
```

3.2 Acknowledgement of non-candidate coding

All the code was written by me. No chunks of code were copied.



Testing

Fluid dynamics simulator

This is the testing section of my coursework. In this section I will be testing the functionality of my program. This will include black box and white box testing. The screenshots of the tests can be found in the appendices.

Victor Sim
3/26/2015

4. Testing

4.1 Test plan

I am going to do Top-down testing. Top-down testing is also known as interface testing and stub testing. Basically a skeleton of the complete system is tested. I can try testing the whole functionality of the system by clicking on the buttons to see whether it takes me to a different page and moving the sliders to see if it moves and changes the value of a variable etc. Once I make sure the user interface is working, and then I can correct the details of the program e.g. I click on a button but it doesn't bring me to the right page so I make it bring me to the right page. After the user is able to interact with the system I can see the outputs it produces. Now I can test the lower level modules. After letting the simulation run I will be able to compare the outputs it produces with the actual values I work out by hand. I work downwards through the higher level modules to the lower level modules. If there is an error in my outputs which there were last time, I will check the top one first. If there is no issue with that, then I go onto the subroutines or functions called inside it and check whether they're working properly. I will also check the parameters passed onto the different modules and check if they are correct.

4.2 Test data

Main menu

The following tests are examples of black box testing

Purpose of tests:

1. Check whether the buttons are working correctly
2. Check whether there are sounds heard when a button is clicked

Test no.	Description of test	Test data	Expected result	Actual result	Evidence in appendix
1	I will click on each of the buttons in the main menu and see if it goes to the right pages.	**Typical data:** Click on the simulator button	**Typical data:** It will go to the simulator page	**Typical data:** It went to the simulator page	From Fig 1.0 to Fig 1.1
		Typical data: Click on the Quick quiz button	**Typical data:** It will go to the quick quiz	**Typical data:** It went to the quick quiz	From Fig 1.0 to Fig 1.2
		Typical data: Click on the Calculation quiz button	**Typical data:** It will go to the calculation quiz	**Typical data:** It went to the calculation quiz	From Fig 1.0 to Fig 1.3
2	I will click on each of the buttons to see if a sound effect is heard	**Typical data:** Click on the simulator button	**Typical data:** Sound effect will be heard	**Typical data:** Sound effect was heard	N/A

		Typical data: Click on the Quick quiz button	**Typical data:** Sound effect will be heard	**Typical data:** Sound effect was heard	N/A
		Typical data: Click on the Calculation quiz button	**Typical data:** Sound effect will be heard	**Typical data:** Sound effect was heard	N/A

Simulator menu

Test no. 2 and especially test no. 13 are the main sources of white box testing. The rest were mainly black box testing. Test no. 9 was my interface testing where I made sure that all the links were taking me to the right pages.

Purpose of tests:

1. To test whether the ball with the lowest mass will sink or not
2. To test whether the simulator displays the right output of weight and volume on screen when the properties of the ball is changed
3. To test whether the program catches the fact that there is no fluid.
4. If there is a fluid then the colour of the fluid should appear or change and the output for the fluid density and fluid viscosity would change as well.
5. To see whether changing the radius using the scroll bar would change the radius of the ball in real time
6. To see whether the size of the ball and the picture of the ball would change as I selected a ball
7. To check whether the simulation runs when the run button is clicked
8. To check whether the pause button is working
9. To check if the main menu link is working
10. To test if the ball stops moving once it has reached the bottom of the tube
11. To test if the program stops running once I close the window
12. To test if the program prompts for the location of where to output the simulation data to
13. To test if the simulator produces all the right outputs while the simulation is running

Test No.	Description of the test	Test data	Expected result	Actual result	Evidence in appendix
1	I will set the radius of the ball to the smallest possible i.e. 2mm and the density of the ball to be the lowest i.e. $2gcm^{-3}$. I will then choose all the fluids individually to	**Extreme data:** For all fluids: Radius: 2mm, density: $2gcm^{-3}$	**Extreme data:** The ball will sink	**Extreme data:** The ball sank	2.10 2.11 2.12 2.13 2.14 2.15 2.16

	check whether it would sink for all of them.				2.17
2	I will change the radius and the density of the ball using the scroll sliders and by selecting the ball from the drop down menu. It should produce the correct weight and volume.	**Typical data:** Radius: 5mm, Density: 5gcm^{-3}	**Typical data:** Weight: 0.02568N Volume: 5.236x10^{-7}m^{3}	**Typical data:** Weight: 0.02568N Volume: 5.236x10^{-7}m^{3}	2.18
		Extreme data: Radius: 8mm Density: 10 gcm^{-3}	**Extreme data:** Weight: 0.21039N Volume: 2.1447x10^{-6}m^{3}	**Extreme data:** Weight: 0.21039N Volume: 2.1447x10^{-6}m^{3}	2.19
3	I will try running the simulation with no fluid selected yet. This should display a warning message telling the user that no fluid is selected and one needs to be selected.	**Erroneous data:** Fluid select: Blank	**Erroneous data:** Warning message: "No fluid chosen!"	**Erroneous data:** Warning message: "No fluid chosen!"	2.20
4	I will try running the program with a fluid selected. The test tube should now be filled with a coloured fluid with the backcolour of the picture box (the ball) matching the fluid colour.	**Typical data:** Glycerol	**Typical data:** Colour of fluid changes grey, fluid density = 1.26gcm^{-3}, fluid viscosity = 1.412Pas	**Typical data:** Colour of fluid changes grey, fluid density = 1.26gcm^{-3}, fluid viscosity = 1.412Pas	2.21
		Typical data: Sunflower oil	**Typical data:** Colour of fluid changes yellow, fluid density = 0.93gcm^{-3} , fluid viscosity = 0.4914 Pas	**Typical data:** Colour of fluid changes yellow, fluid density = 0.93gcm^{-3} , fluid viscosity = 0.4914 Pas	2.22
		Typical data: Coconut oil	**Typical data:** Colour of fluid changes yellow-green, fluid density = 0.924 gcm^{-3}, fluid viscosity = 0.06Pas	**Typical data:** Colour of fluid changes yellow-green, fluid density = 0.924 gcm^{-3}, fluid viscosity = 0.06Pas	2.23
		Typical data: Olive oil	**Typical data:** Colour of fluid changes dark green, fluid density = 0.86 gcm^{-3}, fluid viscosity = 0.1075Pas	**Typical data:** Colour of fluid changes dark green, fluid density = 0.86 gcm^{-3}, fluid viscosity = 0.1075Pas	2.24
5	I will move the scroll bar	**Typical data:**	**Typical data:**	**Typical data:**	2.25

	for the radius around. The size of the ball should increase/decrease and the centre of the picture (ball) should remain where it is (so it is aligned in the centre).	Move the slider	As I move the slider to the right (increase the radius) the size of the ball increases, as I move the slider to the left (decrease the radius) the size of the ball decreases. Both of these happen whilst the centre of the ball stays in the centre i.e. the x,y position of the ball doesn't change.	The size of the ball increased as I move the slider to the right and the size of the ball decreased as I moved the slider to the left. The centre of the ball remained at the centre.	2.26
6	I will select each of the balls in the drop down menu and the size should change according to the radius. The picture will also change as the ball is made out of a different material with a specific density.	**Typical data:** 8mm Steel ball	**Typical data:** The size of the ball would change to 8mm. The picture would change into a steel ball.	**Typical data:** The size changed to 8mm and the picture of the ball became a steel ball.	2.27
		Typical data: 8mm Marble ball	**Typical data:** The size of the ball would change to 8mm. The picture would change into a marble ball.	**Typical data:** The size changed to 8mm and the picture of the ball became a marble ball.	2.28
		Typical data: 6mm Steel ball	**Typical data:** The size of the ball would change to 6mm. The picture would change into a steel ball.	**Typical data:** The size changed to 6mm and the picture of the ball became a steel ball.	2.29
		Typical data: 6mm Marble ball	**Typical data:** The size of the ball would change to 6mm. The picture would change into a marble ball.	**Typical data:** The size changed to 6mm and the picture of the ball became a marble ball.	2.30

7	I will click on the run button and observe the ball moving through the fluid if it is working properly.	**Typical data:** All data inputs entered correctly allowing simulation to run	**Typical data:** The ball would start moving downwards	**Typical data:** The ball did start moving downwards	2.10 2.11 (Same as in test 1)
		Erroneous data: There isn't a fluid chosen yet	**Erroneous data:** An error message will be displayed saying "No fluid is chosen!"	**Erroneous data:** An error message is outputted when there is no fluid chosen	2.20 (same as test 3)
8	I will click on the pause button while the simulation is running. This should pause the simulation until the run button is clicked again	**Typical data:** Pause button clicked while simulation running	**Typical data:** The ball would stop moving, the time and other values such as the velocity would stop updating	**Typical data:** The ball stopped moving, the time and the other values stopped updating	2.31
		Extreme data: Pause button clicked while simulation has not started	**Extreme data:** Nothing happens	**Extreme data:** Nothing happened	N/A
9	This is to check whether the user can go back to the main menu to access the different modes.	**Typical data:** Main menu link clicked in simulation mode	**Typical data:** Brings the user back to the main menu	**Typical data:** Brought the user back to the main menu	2.32 2.33 2.34 Going from main menu to simulator back, to main menu,
		Typical data: Main menu link clicked in quick quiz mode	**Typical data:** Brings the user back to the main menu	**Typical data:** Brought the user back to the main menu	2.35 2.36 to quick quiz, back to main menu

		Typical data: Main menu link clicked in calculation quiz mode	**Typical data:** Brings the user back to the main menu	**Typical data:** Brought the user back to the main menu	2.37 2.38 then to calculation quiz and finally back to the main menu
10	I will run the simulation. Once the ball has reached the bottom of the tube, the ball should stop moving and the timer disabled	**Typical data:** Run the simulation	**Typical data:** The ball would touch the bottom of the tube and stop moving	**Typical data** The ball touched the bottom of the tube and stopped moving	2.39
11	I will close the window of the program while it is open on any form. This is to check it closes the other hidden forms with it so that it doesn't just disappear with it still running in the background.	**Typical data:** Close the window on any form	**Typical data:** The entire program will close	**Typical data:** The entire program closed	N/A
12	I will run a simulation and wait for it to stop. The export data button should now appear giving me a chance to export the data. Upon clicking the button, it will display an input box for me to input the file path and the name which I want to call the document	**Typical data:** C:\Users\Victor\ Documents\Test \data.csv	**Typical data:** This would create a CSV file in that location called data	**Typical data:** This created a CSV file in that location called data	For typical, erroneous and extreme data 2.39 (same as in test 10 with export button showing up) **Typical data:** 2.40

		Erroneous data: 'Blank space'	**Erroneous data:** Produce an error message saying 'Invalid file location/File is open'	**Erroneous data:** Produces an error message saying 'Invalid file location/File is open'	**Erroneous data:** 2.41
		Extreme data: abcdefg	**Extreme data:** Creates a file called abcdefg in the AppData of the computer	**Extreme data:** Created a file called abcdefg in the AppData of the computer	**Extreme data:** 2.42
		Extreme data: C:\Users\Victor\Documents\Test\data	**Extreme data:** Creates a file called data in the specified location but it is not of a csv format so the user cannot open it	**Extreme data:** Created a file called data in the correct specified location but is not of the correct format so the user can't open it	**Extreme data:** 2.43
13		**Typical data:** Radius: 6mm Density: 5gcm^{-3} Fluid: Glycerol (density: 1.26g cm^{-3}, viscosity:1.412Pas)	**Typical data:** The outputs such as the current velocity, viscous drag, terminal velocity and up-thrust should match up to the values on my spreadsheet	**Typical data:** The outputs match with the values on the spreadsheet, therefore it is correct	**Typical data:** 2.44
		Extreme data: Radius: 8mm Density: 10gcm^{-3} Fluid: Coconut oil(density: 0.924g/cm^3, viscosity: 0.06 Pas)	**Extreme data:** The outputs such as the current velocity, viscous drag, terminal velocity and up-thrust should match up to the values on my spreadsheet	**Extreme data:** The outputs match with the values on the spreadsheet, therefore it is correct	**Extreme data:** 2.45

		Extreme data: Radius: 8mm Density: 10gcm^{-3} Fluid: Olive oil(density: 0.86g/cm^{3}, viscosity: 0.1075 Pas)	**Extreme data:** The outputs such as the current velocity, viscous drag, terminal velocity and up-thrust should match up to the values on my spreadsheet	**Extreme data:** The outputs match with the values on the spreadsheet, therefore it is correct	**Extreme data:** 2.46
		Extreme data: Radius: 2mm Density: 2gcm^{-3} Fluid: Glycerol(density : 1.26gcm^{-3}, viscosity: 1.412Pas)	**Extreme data:** The outputs such as the current velocity, viscous drag, terminal velocity and up-thrust should match up to the values on my spreadsheet	**Extreme data:** The outputs match with the values on the spreadsheet, therefore it is correct	**Extreme data:** 2.47
		Extreme data: Radius: 8mm Density: 8.05gcm^{-3} Fluid: Sunflower oil(density: 0.93 gcm^{-3}, viscosity: 0.4914)	**Extreme data:** The outputs such as the current velocity, viscous drag, terminal velocity and up-thrust should match up to the values on my spreadsheet	**Extreme data:** The outputs match with the values on the spreadsheet, therefore it is correct	**Extreme data:** 2.48

Quiz menu

The tests for the quiz mode are mainly black box testing.

Purpose of tests

1. To see if the quiz outputs the next question if the user gets it right
2. To see if the quiz stops once the user gets a question wrong
3. Check whether the score is updating when you get a question right
4. To check whether the sound effects are being outputted when you click on the answer button

Test No.	Description of the test	Test data	Expected result	Actual result	Evidence in appendix
1	I will try inputting the right answer to a question. After the answer button is clicked, the system should produce a positive sound, erase the textbox and display the next question.	**Typical data:** Correct answer e.g. yes	**Typical data:** Sound heard, textbox is cleared, next random question produced	**Typical data:** Sound heard, textbox is cleared, next random question produced	3.1 3.2 Showing the first question that was displayed then another question
2	I will try inputting a wrong answer. The quiz should disable the answer button; preventing the user to answer any more questions A negative sound should be heard. The label would continue to display the current question and the textbox will not be cleared. The correct answer will also be displayed in a message box and the total score displayed.	**Typical data:** Incorrect answer e.g. turbulent flow	**Typical data:** Sound heard, textbox is not cleared, answer button is disabled, question remains on screen, correct answer to question displayed	**Typical data:** Textbox is not cleared, answer button is disabled, question remains on screen, correct answer to question displayed	3.3 3.4 3.5
3	After inputting the correct answer, the score on the upper left hand corner of the screen should update by an increment of 10	**Typical data:** Correct answer entered	**Typical data:** The score should increase by 10 from its previous score	**Typical data:** The score increased by 10 from its previous score	3.6
4	I will click on the answer button and check to hear the right sound for each type of answer i.e. positive sound for correct	**Typical data:** Correct answer entered	**Typical data:** The system should produce a positive sound	**Typical data:** The system produced a positive sound	N/A

	answer and negative sound for wrong answer	**Typical data:** Wrong answer entered	**Typical data:** The system should produce a negative sound	**Typical data:** The system produced a negative sound	N/A

4.3 Test evaluation

Problems fixed:

Whether it would float or not

The density of the ball must not be lower than the density of the fluid, otherwise it would float.

This is shown by the formula

Resultant force = Weight – (Viscous drag + upthrust)

At t = 0

Resultant force = Weight – upthrust = volume of ball * density of ball – volume of ball * density of fluid

As glycerol was the most dense liquid among the 4 with a density of $1.26gcm^{-1}$, the lowest integer value of the ball is 2. Therefore, when creating the slide bar for adjusting the density the lower range is 2.

The density of glass and steel are $2.4gcm^{-1}$ and $8.05gcm^{-1}$ respectively which are both greater than 2 so pose no problem.

However, that is with the initial conditions at t = 0. As time increases, velocity increases and so would viscous drag. It is possible that the viscous drag will be big enough and combine with the upthrust to produce a negative velocity; which we don't want in our simulation.

To overcome this problem, the minimum radius which the user can choose from is 2mm.

Strengths in testing

After doing the testing of my system, I checked that most of my functionality and my entire user interface were working. There were slight errors like which I found out during my testing like the ball not staying in the same place when I changed the radius of the ball which was fixed by doing some calculations. All the buttons in the user interface like the run button or the main menu button were all working so the user can navigate through and use the program easily. The functionality of my program was mostly working as the simulator produced all the right outputs of the different variables (e.g. current velocity) of the simulation. During the implementation phase however, there were a lot of values which I needed to fix. Some of these were due to forgetting to convert them to SI units or forgetting to multiply by the Earth's gravitational field strength etc.

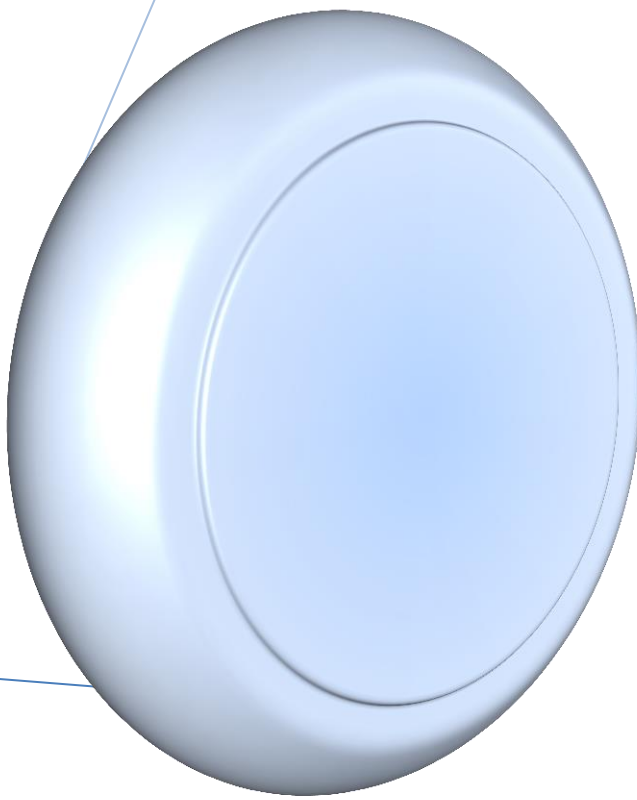
Weaknesses in testing

There were some major errors in my testing which I had to fix. For example, when I compiled the program and tried to run it on another computer, I realised that it cannot read in the bitmap, csv and wav files because it was still trying to read them in from a local location. This forced me to store them as resources in VB so it can be stored in the executable file. After that, the program was able to work on any computer. Another problem I faced was that I wasn't sure how to have the csv file stored on someone's computer and have my program read it in. I wanted the user to have access to the csv file so if my client (the teacher) decides to change any of the questions or add any more she can do so. This allowed me to look up `My.Computer.FileSystem.SpecialDirectories.MyDocuments` which allows me to get the location of each unique individual location.

I also found out that while exporting another set of data right after I exported one set of data, the second set didn't export properly. Then I realised that I didn't initialise all my variables again, setting them to 0. After fixing this problem, I could export the data without any issues.

Beta testing

I have also done some beta testing. This can be found in my appraisal section where I asked the students to fill out a questionnaire including seeing if they feel the system meets all the objectives and what they thought could be improved.



System maintenance

Fluid dynamics simulator

In this section of the coursework, I will be going through the system overview in greater detail. I will also compare the pseudocode from the design section with the actual code. There would be a labelled diagram of all the objects on the form, explaining what each of them does. The code listing can be found in the appendix.

Victor Sim
3/26/2015

5. System maintenance

5.1 System overview

I have created a system containing four forms. They contain the MainMenu, SimulatorProgram, QuickQuiz and CalculationQuiz. First I am going to talk about my main form that is the MainMenu class. This form allows the user to navigate through the different forms which includes the SimulatorProgram, QuickQuiz and CalculationQuiz. This form also displays the title of the program i.e. "Physics Fluid Dynamics Simulator". To get back to the main menu form, there is a hyperlink on each of the other forms which the user can click on.

The form which contains most work and is the main objective for this project is the SimulatorProgram form. On this form, the user is able to experiment with the radius and density of the ball as well as the different types of fluid to see how this would affect the variables such as the terminal velocity of the ball. On this form, the user can click on the run and pause button to start or pause the simulation after a fluid has been selected. A fluid is selected using a drop down menu with the choices glycerol, sunflower oil, coconut oil and olive oil. After selecting a fluid, the picture of the tube would change. A pre-defined ball with a set radius and density can also be selected from a drop down menu which contains the choices of 8mm steel ball, 6mm steel ball, 8mm marble ball and 6mm marble ball. After selecting a ball, the picture of the ball would change both in size and picture. If the user wants to create their own ball, they can use the sliders provided to change the radius or density of the ball. The size of the ball would change as the user is moving the slider to alter the radius. After running the simulation, the variables of the simulation are updated on the left side of the screen by labels. Finally, after the simulation has ended, an "Export data" button will appear giving the user an option of exporting their data from the run to a CSV file. When the simulation runs again, the export data button disappears until the end of the simulation is reached.

My QuickQuiz form and CalculationQuiz form are essentially the same thing. They both contain a label on the top left keeping track of the user's score, a question to be displayed using a label, the title of the form and a text box with an answer button beside it to input their answers in. The main difference between these two forms is the type of questions which they are presenting, the QuickQuiz presenting yes/no type of questions whereas the CalculationQuiz involves calculations. They read in different CSV files from where they get the questions.

5.2 Sample of algorithms

The following pseudocode can be found in the design section but here it is again.

Pseudocode	**Actual code**

Calculate volume
BEGIN
'radius in units of mm
Volume = (4*pi*(radius/1000) ^3)/3
END
	`Public Sub CalculateVolume(radius As Integer)`
	`'Radius is in units of mm`
	`p_volume = (4 * System.Math.PI * (radius / 1000) ^ 3) / 3`
	`End Sub`

Calculate mass BEGIN 'Have to multiply the density by 1000	`Public Sub Calculatemass(volume, density)` `'density is in g/cm^3 so have to convert it to kg/m^3 by multiplying by 1000`

to get it from gcm^{-3} to kgm^{-3} Mass = volume * density * 1000 END	p_mass = volume * density * 1000 End Sub
Calculate weight BEGIN 'Calculates the weight using mass Weight = mass*g END	Public Function Getweight() 'Calculates the weight using mass p_weight = p_mass * g Return p_weight End Function
Calculate viscous drag BEGIN 'Calculates viscous drag acting on ball using Stokes' law equation ViscousDrag = 6*pi*v*r* η END	Public Sub CalculateViscousDrag(radius, viscosity) 'Calculates the viscous drag acting on the ball using Stokes' law p_ViscousDrag = 6 * System.Math.PI * p_CurrentVelocity * radius * viscosity End Sub
Calculate up-thrust BEGIN 'Denisty of fluid in gcm^{-3} so have to convert it to kgm^{-3} Upthrust = volume * DensityOfFluid * 1000 * g END	Public Sub CalculateUpthrust(volume, densityOfFluid) 'Caclulates up-thrust using volume and density 'Multiplied by 1000 to get it in SI units p_Upthrust = volume * densityOfFluid * 1000 * g End Sub
Calculate new (current) velocity BEGIN 'Calculate resultant force using free body force diagram Resultant = Weight – (ViscousDrag + Upthrust) 'Gets acceleration from Newton's second law Acceleration = Resultant/Mass 'Gets new velocity using SUVAT v = u + at 'Updates with new velocity u = v END	Public Sub CalculateAcceleration(Weight, Mass) 'Calculates the resultant force p_Resultant = Weight - (p_ViscousDrag + p_Upthrust) 'Uses Newton's 2nd law to get the acceleration p_Acceleration = p_Resultant / Mass End Sub Public Sub CalculateNewVelocity() 'Uses the SUVAT equation v = u + at p_CurrentVelocity = p_OldVelocity + p_Acceleration * time 'Updates the old velocity to become the new velocity p_OldVelocity = p_CurrentVelocity End Sub
Calculate distance moved by the ball BEGIN 'Gets distance by doing the instantaneous velocity multiplied by the time interval for each update Distance = currentVelocity * time END	Public Sub CalculateDistance() 'Calculates the distance it should move by using distance = speed * time 'Multiplies the time by 100000 to get noticable movement because time in order of magnitude 10^-3 and velocities are very small p_Distance = p_CurrentVelocity * time * 100000 End Sub
Calculate terminal velocity of the ball through fluid BEGIN $(2*radius^2*g*(densityOfSolid – densityOfFluid))/(9*viscosity)$	Public Sub CalculateTerminalVeloctiy(radius, solidDensity, fluidDensity, viscosity) 'Calculates the terminal velocity using equation (2r^2(ρs – ρf))/(9η) 'Divides and multiplies by 1000 to

END

```
get it into SI units
        p_TerminalVelocity = (2 * (radius / 1000) ^ 2 * g * (solidDensity * 1000 - fluidDensity * 1000)) / (9 * viscosity)
    End Sub
```

Checking whether the ball is going out of tube

BEGIN
    REPEAT
        'Checks if the ball's position + the height/radius of the ball to account for the different radii of the ball exceeds the position of the bottom of the tube
        IF BallPosition + BallHeight > 438 THEN
            Timer = False
        ENDIF
    UNTIL BallPosition + Ball Height >438
END

```
    Private Sub CheckGoingOutOfScreen()
        'Makes sure the ball doesn't go out of the tube
        If myball.Y + myball.image.Height > 438 Then
            'Allows the user to set up the next run
            PrepareForNextRun()
        End If
    End Sub
```

Exporting data

BEGIN
    INPUT FileLocation
    IF FileLocation <> "" THEN
        'Outputs the labels
        OUTPUT "Time (s), Velocity (m/s), Resultant force (N), Viscous drag (N), Acceleration
        (m/s^2), Terminal velocity, m/s, Upthrust, N, Radius, mm, Weight, N, Volume, m^3,
        Density of ball, g/cm^3, Fluid, Fluid density, g/cm^3, Viscosity, Pas"
        'Outputs the pieces of data for the corresponding labels
        For x = 1 to Time*1000 + 1
            OUTPUT data(0, x)
            OUTPUT data(1, x)
            OUTPUT data(2, x)
            OUTPUT data(3, x)
            OUTPUT data(4, x)
            OUTPUT new line
        NEXT
        'Informs the user whether the data exportation was successful or not
        OUTPUT "data exported!"
    ELSE
        OUTPUT "invalid file location"
    ENDIF
END

```
    Private Sub btnExport_Click(sender As System.Object, e As System.EventArgs) Handles btnExport.Click
        'Creates a streamwrite class
        Dim StreamToWrite As StreamWriter
        Dim FileLocation As String

        'User inputs where they want to write the file and the file name
        FileLocation = InputBox("Please type in the file path followed by the file name which you want it to be called followed by .csv")
        If FileLocation <> "" Then
            Try
                StreamToWrite = New StreamWriter(FileLocation)
                'Write the first row of the table
                StreamToWrite.Write("Time (s)" & "," & "Velocity (m/s)" & "," & "Resultant force (N)" & "," & "Viscous drag (N)" & "," & "Acceleration (m/s^2)")
                'Writes all the information that doesn't need to be updated every timer tick
                StreamToWrite.Write(",," & "Terminal velocity" & "," & mySimulator.TerminalVelocity & "," & "m/s" & "," & "Upthrust" & "," & _

mySimulator.Upthrust & "," & "N" & "," & "Radius" & "," & myBall.radius & "," & "mm" & "," & "Weight" & "," & _

myBall.weight & "," & "N" & "," & "Volume" & "," & myBall.volume & "," & "m^3" & "," & "Density of ball" & "," & _
```

	myBall.density & "," & "g/cm^3" & "," & "Fluid" & "," & myFluid.fluid & "," & "Fluid density" & "," & myFluid.density & _ "," & "g/cm^3" & "," & "Viscosity" & "," & myFluid.viscosity & "," & "Pas") 'Creates next row StreamToWrite.Write(vbCrLf) 'Writes the data row by row from the array that stored the values of simulation just now 'The time is multiplied by 1000 because they go in 0.001 second intervals so 0.001*1000 = 1 and + 1 because time starts from 0 For x = 1 To Time * 1000 + 1 StreamToWrite.Write(data(0, x) & ",") StreamToWrite.Write(data(1, x) & ",") StreamToWrite.Write(data(2, x) & ",") StreamToWrite.Write(data(3, x) & ",") StreamToWrite.Write(data(4, x)) StreamToWrite.Write(vbCrLf) Next 'Notifies user data has been exported MsgBox("Data exported!", MsgBoxStyle.Exclamation) 'Closes streamwriter file so can write again next time StreamToWrite.Close() Catch ex As Exception 'Alerts the user that an error has occured MsgBox("Invalid file location/File is open", MsgBoxStyle.Critical) End Try End If End Sub

Reading files from quiz
BEGIN
'Tells the program which file to read in depending on which type of quiz the user selected
INPUT quizType
IF quizType = "Calculate" THEN
READ CalculationQuizFile
ELSE
READ QuickQuizFile
ENDIF
'Reads in each question from the file. A line being one question.
	Public Sub ReadFile(quiztype)
	'Creates streamreader class
	Dim fileReader As StreamReader
	'Reads the short questions for QuickQuiz by default
	fileReader = New StreamReader(My.Computer.FileSystem.SpecialDirectories.MyDocuments & "\Short questions.csv")
	'If it is the CalculationQuiz thenr reads in the calculation questions
	If quiztype = "calculate" Then
	fileReader = New StreamReader(My.Computer.FileSystem.SpecialDirectories.MyDocuments & "\Long questions.csv")

REPEAT Questions(x) = ReadLine x = x+1 UNTIL EndOfFileReached END	End If 'Stores each line of the file in an array until end of file is reached Do Questions(x) = fileReader.ReadLine() x += 1 Loop Until fileReader.EndOfStream 'Closes filereader so it can read files again later fileReader.Close() 'Redims the Questions array which was initially Questions(50) to how many questions there actually are on the document to save memory space ReDim Preserve Questions(x) End Sub

CheckingAnswer
BEGIN
'Keeps checking whether the user inputted the correct answer or not
REPEAT
INPUT answer
IF answer = storedAnswer THEN
OUTPUT positiveSound
score = score + 10
ENDIF
UNTIL answer <> storedAnswer OR questionsAnswered = noOfQuestions
IF answer <> storedAnswer
'Once the user gets the question wrong, the system will output a negative sound, inform the user of the correct answer and their final score
OUTPUT negativeSound
OUTPUT storedAnswer
OUTPUT score
ELSE
'If the user manages to do all the questions correctly, then a congratulations message will appear then output their score which will also be the maximum possible score
OUTPUT "Well done you finished the game!"
OUTPUT score
END
	Public Sub CheckAnswer(ByVal Answer As String, ByRef Correct As Boolean)
	'Converts the use input to capital letters so that it is not case sensitive
	'Checks user's answer with answer part of the line
	If UCase(Answer) = Mid(Questions(questionNo), z + 2, Len(Questions(questionNo)) - z + 2) Then
	'Executes these lines if the user got it right
	'Plays sound effect indicating the user got it right
	
	audioDevice.Play(My.Resources.Correct, AudioPlayMode.Background)
	'Updates score
	p_score += 10
	Else
	'Execute these lines if the user got it wrong
	'Plays a sound effect indicating the user got it wrong
	
	audioDevice.Play(My.Resources.Incorrect, AudioPlayMode.Background)
	Correct = False
	'Outputs the correct answer for user's knowledge
	MsgBox("Sorry you got that wrong! The correct answer was: " & Mid(Questions(questionNo), z + 2, Len(Questions(questionNo)) - z + 2))
	'Outputs their score and a message depending on how high their score is
	If Score > 50 Then
	MsgBox("You got a score of: " & p_score & ". Well done!")
	ElseIf Score > 0 Then
	MsgBox("You got a score of: " & p_score & ". Good effort!")
	Else
	MsgBox("You got a score of: " & p_score & ". Need to work harder!")

	End If
	End If
	End Sub
	
	Function GenerateQuestion()
	……
	……
	'x = 0 when all questions answered
	If x = 0 Then
	MsgBox("Well done you finished the game!")
	MsgBox("You got a score of: " & p_score & ". Well done!")
	Return Nothing
	Else
	……
	……
	End If
	End Function

Outputting question for the quiz
BEGIN
'Sets the initial position to look for the comma indicating when the question starts
firstCommaPos = 1
'Get a question number between 1 and the number of questions
QuestionNo = Rand*noOfQuestions + 1
'Finds the position of the first comma indicating the start of a question
REPEAT
IF mid(Questions(QuestionNo), firstCommaPos, 1) = "," THEN
firstPos = true
ELSE
firstCommaPos = firstCommaPos + 1
ENDIF
UNTIL firstPos = true
'Finds the position of the second comma indicating the start of the answer and end of the question
secondCommaPos = firstCommaPos + 2
REPEAT
IF mid(Questions(QuestionNo), secondCommaPos, 1) = "," THEN
secondPos = true
ELSE
secondPos = secondPos + 1
	Function GenerateQuestion()
	Dim firstpos As Boolean = False
	Dim secondpos As Boolean = False
	Dim y As Integer
	'x = 0 when all questions answered
	If x = 0 Then
	MsgBox("Well done you finished the game!")
	MsgBox("You got a score of: " & p_score & ". Well done!")
	Return Nothing
	Else
	y = 1
	'Chooses a random question
	questionNo = CInt(Int(Rnd() * x))
	'Gets question part of the csv file
	Do
	If Mid(Questions(questionNo), y, 1) = "," Then
	firstpos = True
	Else
	y += 1
	End If
	Loop Until firstpos = True
	z = y + 2
	Do
	If Mid(Questions(questionNo), z, 1) = "," Then
	secondpos = True
	Else
	z += 1
	End If
	Loop Until secondpos = True
	'Outputs the question
	Return Mid(Questions(questionNo), y + 2, z - (y + 2))
	End If
	End Function

ENDIF UNTIL secondPos = true OUTPUT mid(Questions(questionNo), firstCommaPos + 2, secondCommaPos - (firstCommaPos +2)) END	

The pseudocode which requires more logical thinking is the extraction of the question and answer in each record for the quiz. This is because you have to make use of the first comma position and the second comma position to extract the question. A lot of thinking has also gone in the movement of the ball using many physics laws such as Stokes' law and Newton's second law. The route to the solution wasn't obvious as you had to link all parts of what you know to simulate the motion until you get to terminal velocity which wasn't taught in class. If just simulating the motion at terminal velocity that would be easy. In the test evaluation of the testing section, I talked about getting some wrong outputs during the simulation. This required me to go back and check my code and one of the reasons was forgetting to convert the units into SI units because all my calculations are meant to be in SI units, if not they would be a scale factor of 10 to the power of something out. Therefore, I annotated my code with why I multiplied something by 1000 or divide it by 1000. The distance the ball should move by at a certain velocity was also hard to gauge. If I wanted the ball to move faster, then I can just multiply the distance it is to move by a larger number. If I wanted the ball to move slower, then I would multiply it by a lower number. I annotated this in my code so other programmers know why I multiplied it by 100000.

5.3 Annotated code listing

The full code can be found in the appendix pages 161 – 178.

5.4 Procedures

The list and explanation of procedures used in the classes can be found in the analysis section

In MainMenu form the subroutines are as follows:

Procedure name	Parameters	Description
Form1_FormClosed	Sender, e	Closes the program
Shutdown	None	Closes all the other hidden forms
Button1_Click	Sender, e	Shows the SimulatorProgram form and hides the current (MainMenu) form when btnSimulator is clicked
Button2_Click	Sender, e	Shows the QuickQuiz form and hides the current form when btnQuickQuiz is clicked
Button3_Click	Sender, e	Shows the CalculationQuiz form and hides the current form when btnCalculationQuiz is clicked
Button_Clicked	Sender, e	Plays a sound effect when a

		button is clicked to emphasise the button click
Form1_Load	Sender, e	Sets the backcolour of the form as beige

In SimulatorProgram form the subroutines are as follows:

Procedure name	Parameters	Description
Form2_FormClosed	Sender, e	Closes the program
Shutdown	None	Closes all the other forms which are open
Form2_Load	Sender, e	Initialises components
FormInit	None	Loads images of tube and ball. Sets the ball's vertical position and size. Loads a backcolour beige for form
LoadDefaultBall	None	Sets radius to 2, sets density to 2. Outputs the value of them. Calculates volume and mass and outputting them as well
LinkLabel1_LinkClicked	Sender, e	Takes the user back to the main menu
TrackBar1_Scroll	Sender, e	Changes the value of the radius of the ball using a slider. The size of the picture of the ball also changes
TrackBar2_Scroll	Sender, e	Changes the density of the ball. Loads another image of the ball to inform the user it is not the steel or marble ball
ChangeSizeOfBall	None	Changes the visual size of the ball so the user can see the difference
btnRun_Click	Sender, e	Checks if a fluid is chosen first. If a fluid is chosen, the timer is enabled and the simulation starts. Disables all the tools to change the ball and fluid as well so that the user cannot change them midway through the simulation
btnPause_Click	Sender, e	Disables the timer and pauses the simulation (animation).
Timer1_Tick	Sender, e	Controls all the things that needed to update every

		timer tick.
UpdateVariables	None	Updates all the output text of the simulation details like the up-thrust, viscous drag etc. Also stores all the values of these variables at the instantaneous time in a 2-D array so the user can export it later
MoveBall	None	Moves the picture of the ball across the screen. Calculates the distance to move the ball by and sets the new position to the old position + distance
CheckGoingOutOfScreen	None	Stops the ball once it reaches the bottom of the tube. Once it reaches the bottom of the tube it sets up for the next run like enabling the radius slider etc.
ChangeSolidVariables	None	Outputs the new values of variables in the labels
PrepareForNextRun	None	Enables all the objects which need to be enabled to set up next run e.g. the scroll slider and disables some of the other objects e.g. the run button until properties involved in simulation changes. Also displays export button to let user export data
cboSelectBall_SelectedIndexChanged	Sender, e	Changes the ball in use. Outputs the new properties of the ball as well
ReadyForNextRun	None	Enables the run button and resets all the variables of the simulation e.g. currentVelocity, TerminalVelocity for the next run
SelectBall	None	Assigns a material to the ball. Assigns the new radius, density and image to myBall of the ball class as well.
cboSelectFluid_SelectedIndexChanged	Sender, e	Assigns myFluid of the fluid class the fluid currently in use and assigns the properties of that fluid to it

SelectFluid	None	Assign the name of fluid to myFluid and changes the properties e.g. density and viscosity of it. Changes the image of the tube to get the right colour for the fluid chosen. Also changes the backColour of the ball to match the colour of the fluid
ChangeFluidVariables	None	Outputs the new properties of the fluid to the user in labels
btnExport_Click	Sender, e	Exports the data which was stored in the 2-D data array into a CSV file. The location and name of the CSV file is specified by the user when an input box is presented to them

In QuickQuiz/CalculationQuiz form the subroutines are as follows:

Procedure name	Parameters	Description
Form2_FormClosed/Form3_FormClosed	Sender, e	Closes the program
Shutdown	None	Closes all the other hidden forms that are open
Form2_Load/Form3_Load	Sender, e	Initialises random number generator so each time program is run a set of different questions will appear. Calls procedure to read in appropriate file of questions and answers depending on whether it is the QuickQuiz or CalculationQuiz. Then it outputs the first question by calling the procedure to output the question. Also sets the backcolour of form to beige
OutputQuestion	None	Makes the text of the question label = the random question generation using the myQuiz object
LinkLabel1_LinkClicked	Sender, e	Hides the current form and brings the user back to the main menu
btnAnswer_Click	Sender, e	Checks the answer of the

		user input with the correct answer. If the answers are the same, then the score is updated and the next question is produced. Otherwise, the final score of the user is outputted and the answer button is disabled thus ending the quiz.
Proceed	None	Updates the array of questions and answers so the same question won't be given twice. Outputs the new question

5.5 Variables

Here is a list of the variables used in my program

In the MainMenu form:

Variable name	Access type	Type	Description
Simulator	Private	SimulatorProgram	Assigns a variable to the SimulatorProgram class to open the form
QuickQuiz	Private	QuickQuiz	Assigns a variable to the QuickQuiz class to open the form
CalculationQuiz	Private	CalculationQuiz	Assigns a variable to the CalculationQuiz class to open the form

In the SimulatorProgram form:

Variable name	Access type	Type	Description
myBall	Private	Ball	Creates an instance of an object of the Ball class
myFluid	Private	Fluid	Creates an instance of an object of the Fluid class
mySimulator	Private	Simulator	Creates an instance of an object of the Simulator class
Time	Private	Single	Keeps track of the time when the simulation is running. It would increase a 0.001

			interval each timer tick
Data(5,999)	Private	String	Stores the data of the simulation to be exported into a CSV file later on
Position	Private	Integer	Stores the value of which row of the data array file it is writing to the CSV file
MainMenu	Private	MainMenu	Creates a new object of the MainMenu class to display
StreamToWrite	Private	Streamwriter	Creates a streamwriter object for writing the data out to an external file
FileLocation	Private	String	Stores the location and file name specified by the user to output the data to

In the QuickQuiz/CalculationQuiz form:

Variable name	Access type	Type	Description
myQuiz	Private	Quiz	Creates an instance of an object of the Quiz class to use the fields and methods in it
MainMenu	Private	MainMenu	Creates an instance of an object of the MainMenu class to display it
Correct	Private	Boolean	Stores whether the user's answer is correct or not. If correct = true allows the user to continue with the quiz, else ends the quiz

Here are the variables in addition to the fields in the classes:

Simulator Class:

Variable name	Access type	Type	Description
g	Private	Const	Stores the value of g (Earth's gravitational field strength) which is 9.81
time	Private	Const	Stores the time

			interval between each calculation

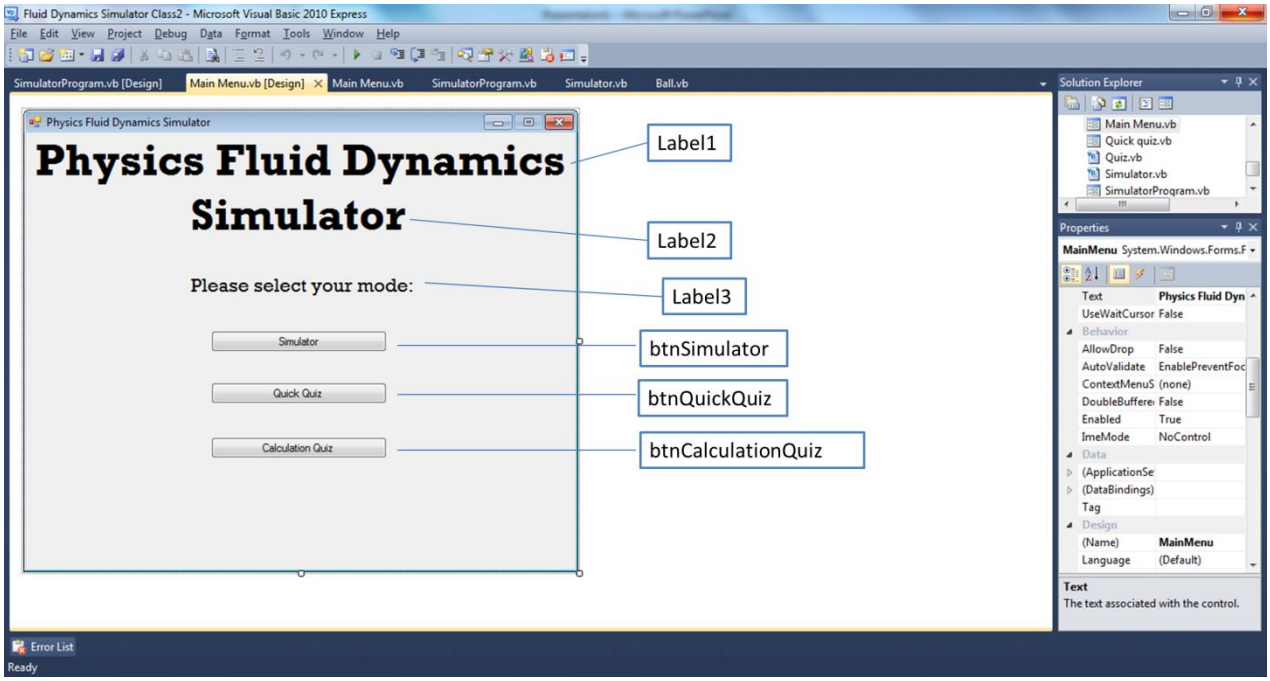
Quiz Class:

Variable name	Access type	Type	Description
z	Private	Integer	Stores the position of the second comma in an element of the data array which separates the question part from the answer part
x	Private	Integer	Stores the current position of the array which is currently being written to
Questions(50)	Private	String	Stores the question numbers, questions and answers from the CSV file
questionNo	Private	Integer	Stores the random question number selected

5.6 Forms

Things used in my MainMenu class (Form1)

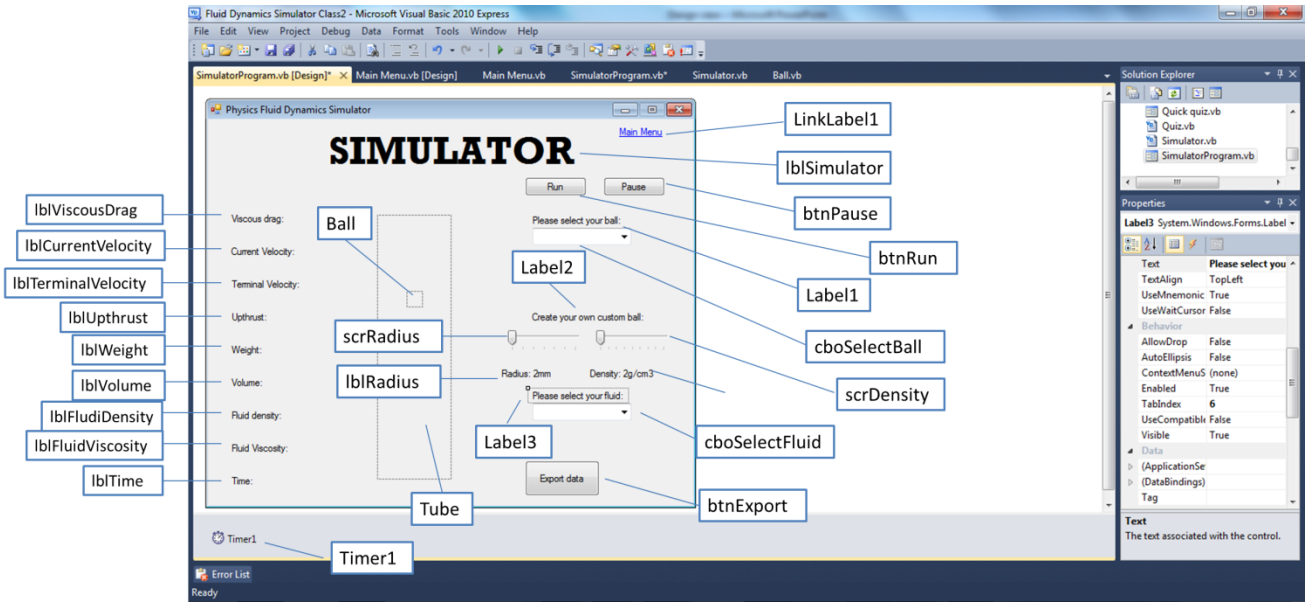
Object	Description
Label1	States the title
Label2	States the title
Label3	Writes a notice to prompt the user to select their desired mode
btnSimulator	Brings the user to the simulator mode once clicked
btnQuickQuiz	Brings the user to the quick quiz mode once clicked
btnCalculationQuiz	Brings the user to the calculation quiz mode once clicked



Things used in my SimulatorProgram class (Form2)

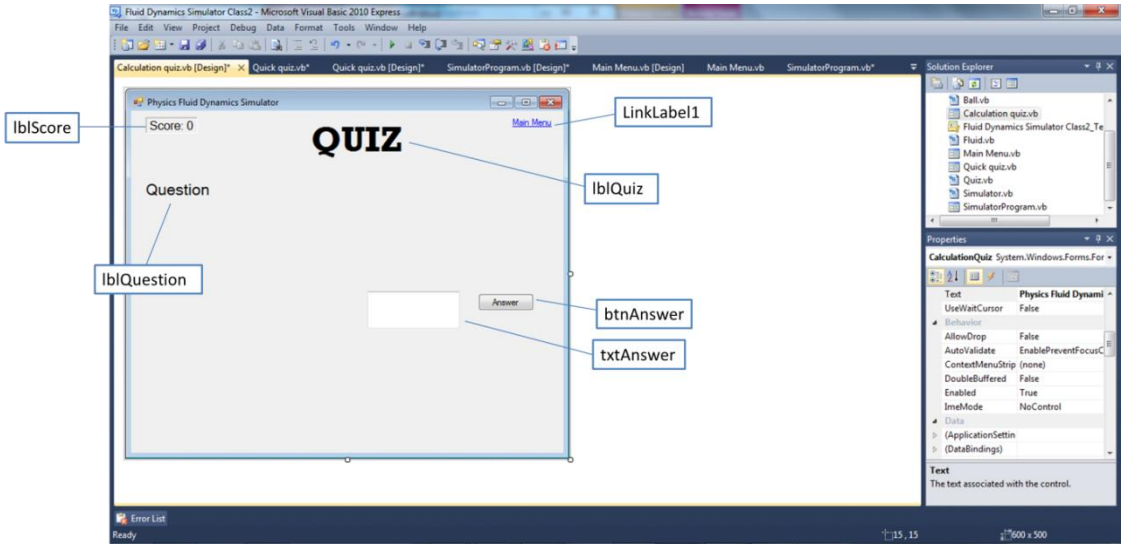
Object	Description
LinkLabel1	Sends the user back to the main menu
lblSimulator	Has the title of “SIMULATOR” for the form
lblViscousDrag	Outputs the viscous drag acting on the ball
lblCurrentVelocity	Outputs the ball’s current velocity
lblTerminalVelocity	Outputs the ball’s terminal velocity
lblUpthrust	Outputs the upthrust acting on the ball
lblWeight	Outputs the weight of the ball
lblVolume	Outputs the volume of the ball
lblFluidDensity	Outputs the density of the fluid
lblFuidViscosity	Outputs the viscosity of the fluid
lbTime	Outputs the duration of the simulation
Tube	Picturebox to load the image of the tube
Ball	Picturebox to load the image of the ball
Timer1	Timer to control the events
btnRun	Runs the simulation
btnPause	Pause the simulation
Label1	Shows text “Please select you ball:”
cboSelectBall	Combo box allowing user to select a pre-defined ball
Label2	Shows text “Create your own custom ball”
scrRadius	Slider allowing user to change the size of the ball
scrDensity	Slider allowing user to change density of the ball
lblRadius	Outputs the radius of the ball which is the value on the slider
lblDensity	Outputs the density of the ball which is the value on the slider
Label3	Shows text “Please select your fluid:”

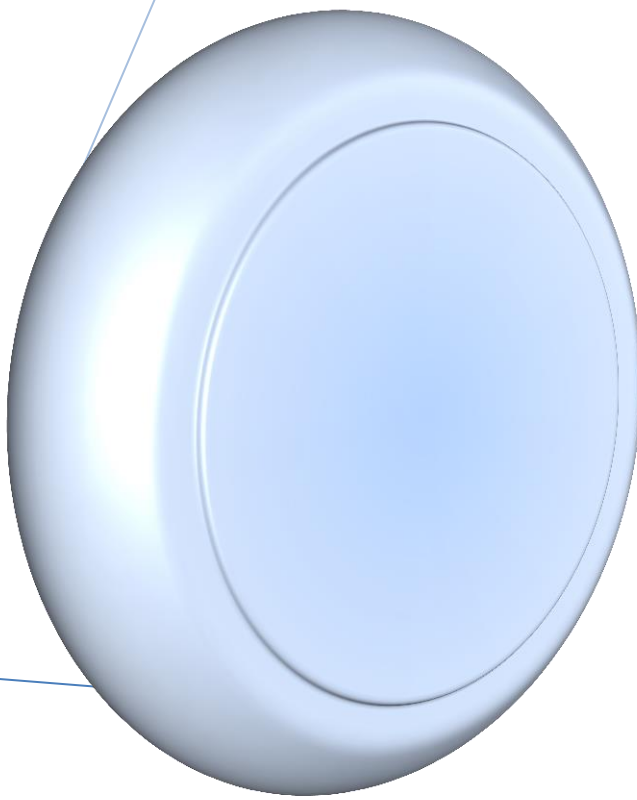
cboSelectFluid	Combo box allowing user to select a fluid
btnExport	Allows user to export data from simulation



Things used in my QuickQuiz and CalculationQuiz class (forms 3 and 4)

Object	Description
lblQuiz	Shows the title of the form
lblScore	Displays the current score of the user
lblQuestion	Displays a random question to the user
LinkLabel1	Allows the user to go back to the main menu
txtAnswer	A textbox which allows the user to enter their answer
btnAnswer	Program checks the user answer after the user clicks the answer button





Appraisal

Fluid dynamics simulator

In the appraisal section of my coursework, I will be talking about how the objectives have been met, my client's and users' feedback and what I can do next time to improve my program.

Victor Sim
3/26/2015

6. Appraisal

6.1 Original Objectives

Objective no.	Objective	Met?	Comments
1	It will make an animation of the ball in fluid	Yes	The ball moves through the fluid as the simulation is run.
2	The ball will stop once it hits the bottom of the container	Yes	The ball stops moving once it has reached the bottom of the container.
3	The user can change the fluid therefore changing the viscosity and density of the fluid	Yes	The user is able to change the fluid with the drop down list. The viscosity and density of the fluid has also changed as you can see the new value in a label.
4	The user can change the ball i.e radius, density etc.	Yes	When the radius slider is moved, the value of the radius changes and therefore the output of the weight and volume changes as well. When the density slider is moved, the value of the density changes and therefore the output of the weight changes as well. When a ball is selected from a drop down list, the properties of the ball such as weight and volume changes as well so the user can most certainly change the ball.
5	The user will be able to know the current speed	Yes	The simulator outputs the current speed when the simulation is run. This value changes every timer tick until terminal velocity is reached.
6	The user will be able to know the terminal velocity	Yes	The terminal velocity is outputted. This matches the current velocity once it stays constant which shows that the current velocity is being updated correctly and that the terminal velocity is correct.
7	The user will be able to know the density of the fluid	Yes	The density of the fluid is displayed as a label on the side. When the user selects a new fluid, the value of the density of the fluid changes.
8	The user will be able to know the volume of the ball	Yes	The volume of the ball is outputted as a label. When the radius of the ball is changed, the volume of the ball is also changed.
9	The user will be able to know the up-thrust acting on the ball	Yes	The user is able to see what the up-thrust is once he has started running the simulation.
10	The colour of the fluid would change as a new fluid is selected	Yes	The colour of the fluid inside the tube as the user selects a new fluid from the drop down menu.

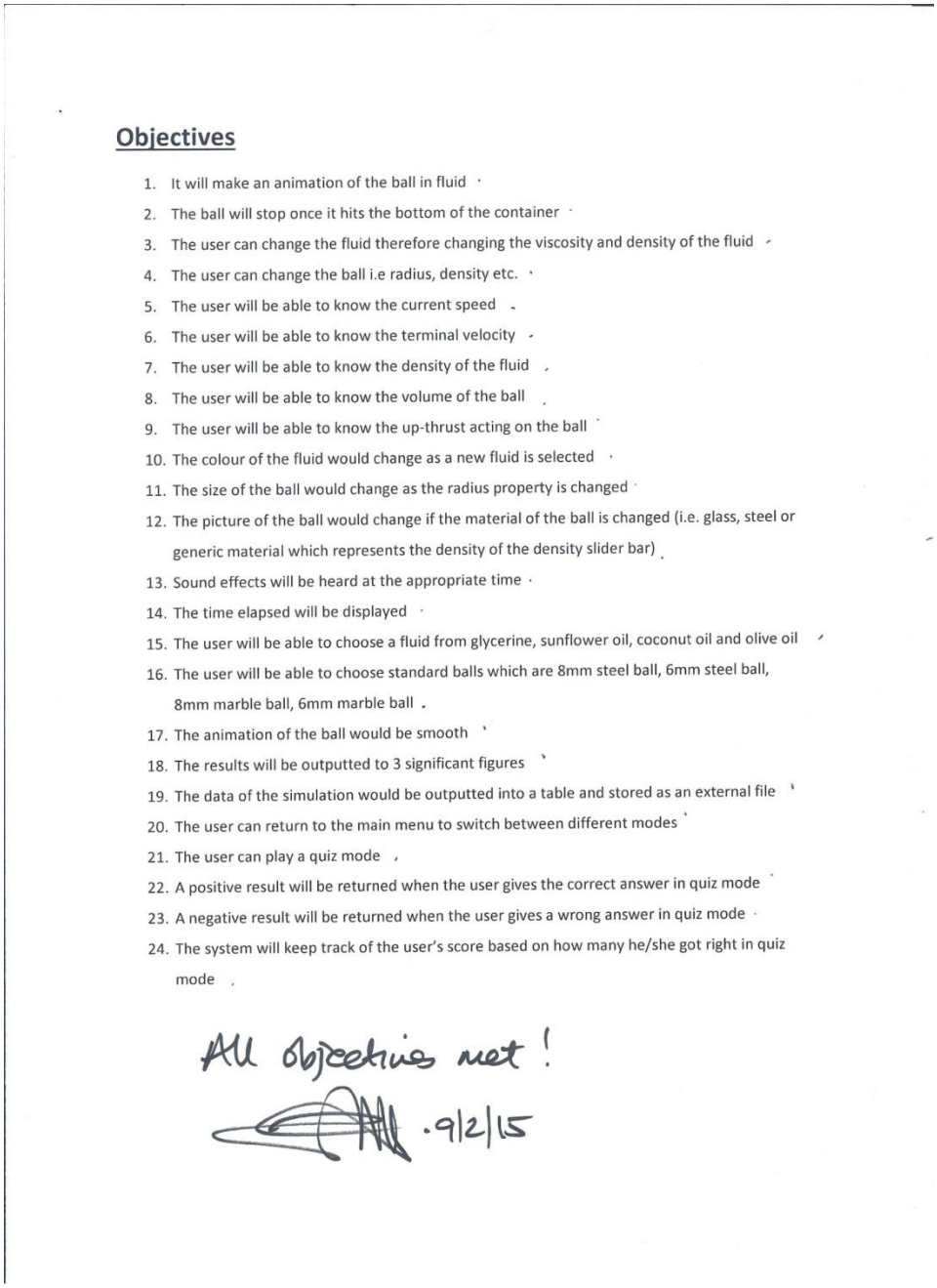
11	The size of the ball would change as the radius property is changed	Yes	The picture varies in size as the radius slider is changed or a new ball is selected from the drop down menu.
12	The picture of the ball would change if the material of the ball is changed (i.e. glass, steel or generic material which represents the density of the density slider bar)	Yes	The picture of the ball changes to a steel ball if the steel ball is selected or the ball changes to a picture of a marble ball if the marble ball is selected. If the user changes the density by using the slider it changes to this blue ball which is supposed to represent any density.
13	Sound effects will be heard at the appropriate time	Yes	When the user clicks on a button on the main menu, a button click sound is heard. There are also sound effects played in the quiz mode when the user gets the answer right or wrong.
14	The time elapsed will be displayed	Yes	The user is able to see the time that has elapsed in simulation time (1 second in real life may represent 0.01 seconds in the simulation) since the simulation has started.
15	The user will be able to choose a fluid from glycerine, sunflower oil, coconut oil and olive oil	Yes	The options of all those fluids are given to the user when the user clicks on the select fluid drop down menu.
16	The user will be able to choose standard balls which are 8mm steel ball, 6mm steel ball, 8mm marble ball, 6mm marble ball	Yes	Those options are given to the user when the user clicks on the select ball drop down menu.
17	The animation of the ball would be smooth	Yes	The ball drops pretty smoothly down the tube.
18	The results will be outputted to 2 - 5 significant figures on the simulation	Yes	The results like terminal velocity, viscous drag are outputted to 2 - 5 significant figures in the labels at the side.
19	The data of the simulation would be outputted into a table and stored as an external file	Yes	The data of the simulation can be stored as a csv file.
20	The user can return to the main menu to switch between different modes	Yes	When the user clicks on the main menu hyperlink on the top right, it brings the user back to the main menu.

6.2 Additional Objectives

Objective no.	Objective	Met?	Comments
21	The user can play a quiz mode	Yes	The user can play a quiz mode by clicking on either the quick quiz button or calculation quiz button in the main menu. This brings them to the appropriate form.

22	A positive result will be returned when the user gives the correct answer in quiz mode	Yes	A positive sound is heard and the score is updated when the user's answer is correct. The next question is then displayed.
23	A negative result will be returned when the user gives a wrong answer in quiz mode	Yes	A negative sound is heard when the user types in the wrong answer. The correct answer is displayed to them as well as the score and the quiz ends.
24	The system will keep track of the user's score based on how many he/she got right in quiz mode	Yes	The score which is displayed in the top left hand corner increments by 10 whenever the user gets a question correct.

The client also agrees that all the objectives have been met:



Objectives

1. It will make an animation of the ball in fluid
2. The ball will stop once it hits the bottom of the container
3. The user can change the fluid therefore changing the viscosity and density of the fluid
4. The user can change the ball i.e radius, density etc.
5. The user will be able to know the current speed
6. The user will be able to know the terminal velocity
7. The user will be able to know the density of the fluid
8. The user will be able to know the volume of the ball
9. The user will be able to know the up-thrust acting on the ball
10. The colour of the fluid would change as a new fluid is selected
11. The size of the ball would change as the radius property is changed
12. The picture of the ball would change if the material of the ball is changed (i.e. glass, steel or generic material which represents the density of the density slider bar)
13. Sound effects will be heard at the appropriate time
14. The time elapsed will be displayed
15. The user will be able to choose a fluid from glycerine, sunflower oil, coconut oil and olive oil
16. The user will be able to choose standard balls which are 8mm steel ball, 6mm steel ball, 8mm marble ball, 6mm marble ball
17. The animation of the ball would be smooth
18. The results will be outputted to 3 significant figures
19. The data of the simulation would be outputted into a table and stored as an external file
20. The user can return to the main menu to switch between different modes
21. The user can play a quiz mode
22. A positive result will be returned when the user gives the correct answer in quiz mode
23. A negative result will be returned when the user gives a wrong answer in quiz mode
24. The system will keep track of the user's score based on how many he/she got right in quiz mode

All objectives met!

9/2/15

6.3 User feedback

I attained the users' feedback by producing a questionnaire (pages 112 – 120 in appendix), asking for the users' opinion on some key features of the program, the usability of the program and the Physics topic I am dealing with in general. The questionnaires can be found in the appendix.

About 60% of the users I've taken find the topic area which my simulation is meant to help in easy, whereas about 40% find it more challenging. My program should help them with the more challenging parts of the topic, for example in questionnaire 1 she/he finds viscous drag rather more difficult. My program would show the relationship between viscous drag and the other variables, e.g. velocity, density or radius. Another good example would be the user who filled questionnaire 5 (page 116) out who finds this topic difficult and the hardest thing is the relationship between terminal velocity and density. Using my program, they will be able to adjust the density and see how this affects the terminal velocity. The quiz mode could also test his/her knowledge on whether the density affects the terminal velocity and if so, how.

Positive user feedback

100% of the users who answered my questionnaire found the user interface to navigate around. This tells me that everything is arranged in a logical manner and is easy to find. All of them also said that they felt like all the objectives have been met (they were given a copy of all the objectives for the new system). This is good as it shows that all the things which my client wants are in the program. 100% of the users who filled out my questionnaire said they will find it useful to have this program to check their experimental results which they got in class with the results produced by the simulation which is one of the original problems in the problem identification section. 90% said they will use this program to aid them with their revision for the exams.

Negative user feedback

There was a lot of criticism in the quiz mode of my program. For example, in questionnaires 2 (pg 113), 5 (pg 116), 7 (pg 118), 8 (pg 119) and 9(pg 120) they all basically said to make a multiple choice quiz so if they input a slightly different answer e.g. "light gate" instead of "light gates" it wouldn't mark them wrong.

6.4 Client feedback

I asked my client to send me an e-mail, telling me what she thinks of the program including the good things about it and what can be improved for next time. A copy of the e-mail can be found in the appendix.

In the e-mail she has briefly outlined the problems with the current system and how the proposed system helps. The e-mail can be found in the appendix page 121 and 122.

Client comments

She said that the she and the students found the simulation easy to use. She used the smartboard to do the quiz as a class and says that the students enjoyed the activity. They were even able to produce a velocity time graph with the exported data which was a kind of bonus as it was not part of the original objectives. By plotting this graph they are able to better understand the motion of the

ball before it hits terminal velocity. I think they could apply this concept to a skydiver who would reach terminal velocity after a while as the graphs have the same shape.

Client criticisms

She felt that the program could have been even better if:

1. The simulation fitted the whole screen and the fluid filled tube was longer
2. A sign indicating that terminal velocity is reached
3. Units for the time
4. Drop down menu for the list of answers possible in the quiz

6.5 Possible extensions

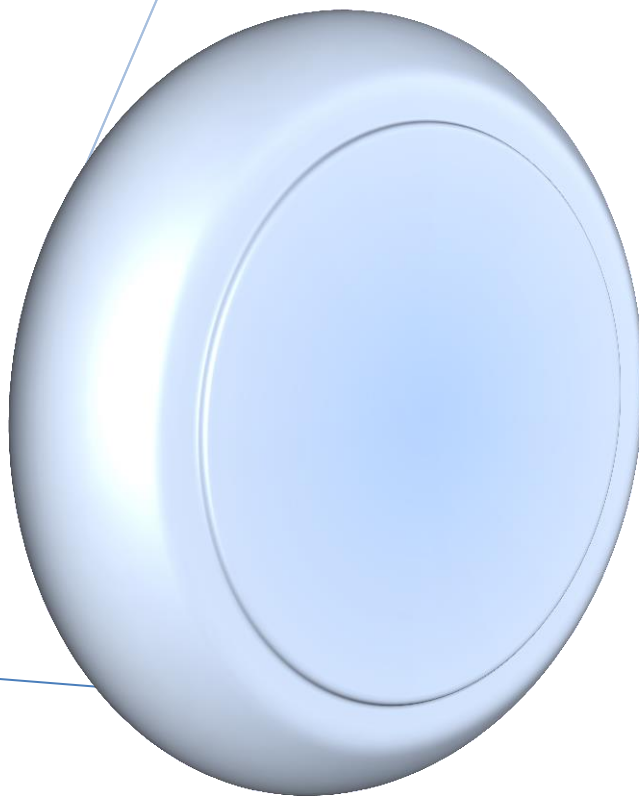
One of the possible extensions I can add to my program is to maybe add one of the things my client and I were discussing about (page 22 of the analysis section): “there will be fluid lines around the ball to show the type of flow (laminar flow)”. This would add to the aesthetics of the program. It would also be quite useful as this brings in more of the laminar/turbulent flow part of the syllabus in and the user can see the important attributes of laminar flow such as the flow lines do not overlap, they are parallel to each other etc. I wouldn’t try to show the flow lines for turbulent flow as modelling the motion of the ball through a fluid that has turbulent flow is beyond my abilities currently. Also, it would defeat the purpose of my client’s needs of wanting to have laminar flow only.

One of the other things I could also try to implement is the encryption of the quiz files which stores the questions and answers to the quiz mode of my program. Right now, if I encrypt the files then the program would fail to read in the files but in the future hopefully the program can read in from encrypted files. The reason for this is to only allow teachers to have access to open the files to modify the questions and prevents students from opening the file and looking up the answers to questions. The teachers will get a password to access the files.

One of the things I can definitely develop for the future from my users’ and client’s feedback is the drop down list for the quiz mode. My client suggested this idea to me in one of our interviews but the reason I didn’t implement it was because I would have to change my entire structure for the quiz mode. The current way my quiz mode works is that it reads in the question and answer from the csv file. If the answer part of the csv file matches what the user wrote, then the system marks it correct. I have already made the answer not case sensitive by making the user’s answer to all upper case letters and checking with the answer which is in upper case. To implement a multiple choice quiz, I would have to replace the textbox with a drop down list containing potential answers to that specific question. This means I will have to change my algorithm of reading in the answer to reading in multiple answers for the choices available in the drop down list. Based on the option the user selected, I can then check whether that option is the correct one by comparing the user’s answer with the real answer.

For the first complaint my client had, I could just make the form size bigger next time which therefore allows me to make the tube longer so more different sized balls would reach terminal velocity. For no.2, I can put up a sign saying that terminal velocity is reached once the current

velocity matches the terminal velocity. I was hoping that the users can see that it has reached terminal velocity themselves once the current velocity output is the same as the terminal velocity output. In no.3, I forgot to put the units of time in my simulation. This caused some confusion as they though that 0.31 was 31 seconds! The terminal velocity is reached very fast and it is important for them to realise this because the real experiment takes about 31 seconds but moves at terminal velocity for most of the time. I was focusing on the first part of the motion so the users can see how the ball moves before it reaches terminal velocity



Appendix

Fluid dynamics simulator

This is the appendix. In this section you can find the interviews with the client, questionnaires from the users as well as the testing screenshots and the code listing

Victor Sim
3/26/2015

7. Appendix

7.1 Questionnaires for analysis

Questionnaire 1

Questionnaire for client for analysis

What variables would you like to alter to find experimenting with these values useful?

Density, radius

What are the outputs you would like to know in the simulation?

Terminal velocity, viscous drag, resultant force

Would you find it useful if you can export the data from the simulation?

☑ Yes

☐ No

How many types of fluid would you like to experiment with?

☐ 1

☐ 2

☐ 3

☑ 4

Do you want to select a ball from a set of pre-defined balls commonly used in laboratory experiments or create a custom ball with your own density and radius?

☐Select a ball

☐Create a custom ball

☑ Both

Questionnaire 2

Questionnaire for client for analysis

What variables would you like to alter to find experimenting with these values useful?

Density, fluid viscosity, fluid density

What are the outputs you would like to know in the simulation?

Current velocity, terminal velocity, time

Would you find it useful if you can export the data from the simulation?

☑ Yes

☐ No

How many types of fluid would you like to experiment with?

☐ 1

☐ 2

☑ 3

☐ 4

Do you want to select a ball from a set of pre-defined balls commonly used in laboratory experiments or create a custom ball with your own density and radius?

☐Select a ball

☐Create a custom ball

☑ Both

Questionnaire 3

Questionnaire for client for analysis

What variables would you like to alter to find experimenting with these values useful?

Radius, material of ball, fluids

What are the outputs you would like to know in the simulation?

Forces acting on ball, e.g. up-thrust. Also properties of ball.

Would you find it useful if you can export the data from the simulation?

☑ Yes

☐ No

How many types of fluid would you like to experiment with?

☐ 1

☐ 2

☐ 3

☑ 4

Do you want to select a ball from a set of pre-defined balls commonly used in laboratory experiments or create a custom ball with your own density and radius?

☐Select a ball

☑ Create a custom ball

☐ Both

Questionnaire 4

Questionnaire for client for analysis

What variables would you like to alter to find experimenting with these values useful?

Viscosity of fluid, density of fluid, density of ball, radius of ball

What are the outputs you would like to know in the simulation?

Terminal velocity, weight, up-thrust, drag

Would you find it useful if you can export the data from the simulation?

☑ Yes

☐ No

How many types of fluid would you like to experiment with?

☐ 1

☐ 2

☐ 3

☑ 4

Do you want to select a ball from a set of pre-defined balls commonly used in laboratory experiments or create a custom ball with your own density and radius?

☐ Select a ball

☐ Create a custom ball

☑ Both

Questionnaire 5

Questionnaire for client for analysis

What variables would you like to alter to find experimenting with these values useful?

Radius

What are the outputs you would like to know in the simulation?

Volume of ball, viscosity of fluid, density of fluid, terminal velocity

Would you find it useful if you can export the data from the simulation?

☑ Yes

☐ No

How many types of fluid would you like to experiment with?

☐ 1

☐ 2

☐ 3

☑ 4

Do you want to select a ball from a set of pre-defined balls commonly used in laboratory experiments or create a custom ball with your own density and radius?

☐ Select a ball

☐ Create a custom ball

☑ Both

7.2 Interviews

Interview 1

Q: What different fluids do they need to have?

A: They need to have glycerine, sunflower oil, coconut oil, olive oil and water. It would be nice if they can select these from a drop down menu.

Q: What are the default balls they need to have?

A: We need 8mm steel ball, 6mm steel ball and a marble ball.

Q: What is the current system?

A: They need to do theoretical calculations themselves and it takes class time away and time from doing practical work.

Q: What will the new system achieve?

A: The new system will allow the students to immediately see what would happen with a different fluid. It must also show the students the terminal velocity, radius of the ball, up-thrust, density of the fluid and current speed. The new system will allow the students to check their experimental data with the values generated by the program.

Q: Do you want a game system in the program?

A: That would be nice but it is not essential. I want the simulation of the ball passing through the fluid first.

Interview 2

Q: So what should be the initial speed of the ball?

A: In a real life experiment the user would just drop the ball into the tube containing the fluid. So in the simulator the ball would just fall through about 5cm of air before hitting the fluid.

Q: Do you want the simulator to display the current speed at all points during its path?

A: That would be nice but what would be more useful is if that it can show the speed at steady intervals. For example, every second or every 1cm it moves through the fluid and storing it in a table.

Interview 3

In this interview, some corrections are made to the program by asking the permission of my client first.

Q: Actually the viscosity and density of water are very low. Is it OK if I take it out? Otherwise, the terminal velocity will be very high.

A: Yes sure.

Q: Also, can I have the ball starting in the fluid first instead of dropping it in? Because it makes the picturebox backcolour easier to deal with that way.

A: Yes you can do that. It doesn't really matter anyway whether you have it starting above the liquid or in the liquid.

Q: There is a problem with reaching terminal velocity for all values the properties of the ball could have. Would you rather limit the range of inputs the user can choose from so that all of them would reach terminal velocity at the end of the simulation or have a range of inputs but not all of them reaching terminal velocity?

A: I would much rather let the students experiment with a wide range of inputs to see how things like radius or different types of fluid affect the movement of the ball. It doesn't really matter if the ball doesn't reach terminal velocity during the simulation because the terminal velocity would be outputted to the user anyways.

Interview 4

These interview questions are more for perfective maintenance.

Q: Do you want a timer for the quiz, or should I just let the students do as many as they can do in their own time until they get something wrong.

A: I would rather let the students do it at their own pace and try get as many questions correct.

Q: How many significant figures do you want the data to be in?

A: 3 significant figures should be fine.

Q: Would you like a separate mode for the slightly longer calculation questions instead of mixing them with the short yes/no questions.

A: Yes that will be nice.

Q: Is it OK if I change the picture of the test tube to this rectangular one, because the box around the image of the ball would protrude out of the test tube if it is curved.

A: Yes that is fine.

Q: Do you want the displacement to be shown in the data outputs of the simulation?

A: No it's OK. The students will be able to work it out anyway from the data you have provided them.

Q: Do you have any more ideas for the questions I could use for the quiz?

A: There are already plenty of questions so I don't think you need any more.

7.3 Questionnaires for appraisal

Questionnaire 1

Questionnaire

For fluid dynamics simulator

Hi, for my computing coursework I need to seek the feedback and approval of the end users that this program is intended for. I will be very grateful if you can answer these few short and easy questions.

1. Did you find the user interface easy to navigate around?

☑ Yes

☐ No

If no, how do you think it can be improved?

---

2. Did you think all the objectives have been met? If not, which ones do you think have not been fully achieved?

Yes

---

3. How do you find the fluid dynamics topic in general? Is it easy or difficult?

Some parts are pretty difficult

---

4. What is the hardest thing you find about this topic?

Viscous drag

---

5. If you had this program to use during class time do you think you would use it to maybe check your experimental results with the calculated results?

☑ Yes

☐ No

6. Do you think you will use this program to aid you in your revision for the exam (e.g. Use the quiz mode to test your knowledge of the topic)?

☑ Yes

☐ No

7. Are there any further improvements you wish to see in the program?

Nah

☺ *Thank you for taking your time to fill out this questionnaire* ☺

Questionnaire 2

Questionnaire

For fluid dynamics simulator

Hi, for my computing coursework I need to seek the feedback and approval of the end users that this program is intended for. I will be very grateful if you can answer these few short and easy questions.

1. Did you find the user interface easy to navigate around?

☑ Yes

☐ No

If no, how do you think it can be improved?

---

2. Did you think all the objectives have been met? If not, which ones do you think have not been fully achieved?

Yes

3. How do you find the fluid dynamics topic in general? Is it easy or difficult?

Easy

4. What is the hardest thing you find about this topic?

Using the equations

5. If you had this program to use during class time do you think you would use it to maybe check your experimental results with the calculated results?

☑ Yes

☐ No

6. Do you think you will use this program to aid you in your revision for the exam (e.g. Use the quiz mode to test your knowledge of the topic)?

☑ Yes

☐ No

7. Are there any further improvements you wish to see in the program?

for the quiz if you don't input the answer exactly correct it says it is wrong e.g. leaving it singular when the correct answer is plural and then it is wrong although the answer is actually correct.

☺ *Thank you for taking your time to fill out this questionnaire* ☺

Questionnaire 3

Questionnaire

For fluid dynamics simulator

Hi, for my computing coursework I need to seek the feedback and approval of the end users that this program is intended for. I will be very grateful if you can answer these few short and easy questions.

1. Did you find the user interface easy to navigate around?

☑ Yes

☐ No

If no, how do you think it can be improved?

---

2. Did you think all the objectives have been met? If not, which ones do you think have not been fully achieved?

---

3. How do you find the fluid dynamics topic in general? Is it easy or difficult?

It is not very difficult

4. What is the hardest thing you find about this topic?

Using the equations in Viscous Drag

5. If you had this program to use during class time do you think you would use it to maybe check your experimental results with the calculated results?

☑ Yes

☐ No

6. Do you think you will use this program to aid you in your revision for the exam (e.g. Use the quiz mode to test your knowledge of the topic)?

☑ Yes

☐ No

7. Are there any further improvements you wish to see in the program?

☺ *Thank you for taking your time to fill out this questionnaire* ☺

Questionnaire 4

Questionnaire

For fluid dynamics simulator

Hi, for my computing coursework I need to seek the feedback and approval of the end users that this program is intended for. I will be very grateful if you can answer these few short and easy questions.

1. Did you find the user interface easy to navigate around?

☑ Yes

☐ No

If no, how do you think it can be improved?

---

2. Did you think all the objectives have been met? If not, which ones do you think have not been fully achieved?

Meets all objectives

---

3. How do you find the fluid dynamics topic in general? Is it easy or difficult?

Medium, average difficulty

---

4. What is the hardest thing you find about this topic?

I find that the explanations hardest

---

5. If you had this program to use during class time do you think you would use it to maybe check your experimental results with the calculated results?

☑ Yes

☐ No

6. Do you think you will use this program to aid you in your revision for the exam (e.g. Use the quiz mode to test your knowledge of the topic)?

☑ Yes

☐ No

7. Are there any further improvements you wish to see in the program?

N/A

☺ *Thank you for taking your time to fill out this questionnaire* ☺

Questionnaire 5

Questionnaire

For fluid dynamics simulator

Hi, for my computing coursework I need to seek the feedback and approval of the end users that this program is intended for. I will be very grateful if you can answer these few short and easy questions.

1. Did you find the user interface easy to navigate around?

☑ Yes

☐ No

If no, how do you think it can be improved?

---

2. Did you think all the objectives have been met? If not, which ones do you think have not been fully achieved?

The objectives have been met.

3. How do you find the fluid dynamics topic in general? Is it easy or difficult?

Difficult.

4. What is the hardest thing you find about this topic?

Relationship between terminal velocity and density.

5. If you had this program to use during class time do you think you would use it to maybe check your experimental results with the calculated results?

☑ Yes.

6. Do you think you will use this program to aid you in your revision for the exam (e.g. Use the quiz mode to test your knowledge of the topic)?

☑ Yes

☐ No

7. Are there any further improvements you wish to see in the program?

Answers that aren't completely correct (spelling) were marked as wrong.

☺ *Thank you for taking your time to fill out this questionnaire* ☺

Questionnaire 6

Questionnaire

For fluid dynamics simulator

Hi, for my computing coursework I need to seek the feedback and approval of the end users that this program is intended for. I will be very grateful if you can answer these few short and easy questions.

1. Did you find the user interface easy to navigate around?

☑ Yes

☐ No

If no, how do you think it can be improved?

---

2. Did you think all the objectives have been met? If not, which ones do you think have not been fully achieved?

Perhaps you could give more generous with the answers for the quiz ☺

3. How do you find the fluid dynamics topic in general? Is it easy or difficult?

Quite straightforward

4. What is the hardest thing you find about this topic?

Remembering the formula for Stoke's Law

5. If you had this program to use during class time do you think you would use it to maybe check your experimental results with the calculated results?

☑ Yes

☐ No

6. Do you think you will use this program to aid you in your revision for the exam (e.g. Use the quiz mode to test your knowledge of the topic)?

☑ Yes

☐ No

7. Are there any further improvements you wish to see in the program?

All objectives have been met! Nice job! ☺

☺ *Thank you for taking your time to fill out this questionnaire* ☺

Questionnaire 7

Questionnaire

For fluid dynamics simulator

Hi, for my computing coursework I need to seek the feedback and approval of the end users that this program is intended for. I will be very grateful if you can answer these few short and easy questions.

1. Did you find the user interface easy to navigate around?

☒ Yes

☐ No

If no, how do you think it can be improved?

---

2. Did you think all the objectives have been met? If not, which ones do you think have not been fully achieved?

Yes everything has been met

3. How do you find the fluid dynamics topic in general? Is it easy or difficult?

Ea Easy to an extent

4. What is the hardest thing you find about this topic?

Viscous drag

5. If you had this program to use during class time do you think you would use it to maybe check your experimental results with the calculated results?

☒ Yes

☐ No

6. Do you think you will use this program to aid you in your revision for the exam (e.g. Use the quiz mode to test your knowledge of the topic)?

☐ Yes

☒ No

7. Are there any further improvements you wish to see in the program?

Include a multiple choice quiz

☺ *Thank you for taking your time to fill out this questionnaire* ☺

Questionnaire 8

Questionnaire

For fluid dynamics simulator

Hi, for my computing coursework I need to seek the feedback and approval of the end users that this program is intended for. I will be very grateful if you can answer these few short and easy questions.

1. Did you find the user interface easy to navigate around?

☑ Yes

☐ No

If no, how do you think it can be improved?

---

2. Did you think all the objectives have been met? If not, which ones do you think have not been fully achieved?

- All objectives achieved

3. How do you find the fluid dynamics topic in general? Is it easy or difficult?

- Some sections easy. Theory and concepts must be understood.

4. What is the hardest thing you find about this topic?

- Viscous drag

5. If you had this program to use during class time do you think you would use it to maybe check your experimental results with the calculated results?

☑ Yes

☐ No

6. Do you think you will use this program to aid you in your revision for the exam (e.g. Use the quiz mode to test your knowledge of the topic)?

☑ Yes

☐ No

7. Are there any further improvements you wish to see in the program?

- Maybe make quiz multiple choice. Some spelling errors in quiz led to wrong answer.

☺ Thank you for taking your time to fill out this questionnaire ☺

Questionnaire 9

Questionnaire

For fluid dynamics simulator

Hi, for my computing coursework I need to seek the feedback and approval of the end users that this program is intended for. I will be very grateful if you can answer these few short and easy questions.

1. Did you find the user interface easy to navigate around?

☑ Yes

☐ No

If no, how do you think it can be improved?

---

2. Did you think all the objectives have been met? If not, which ones do you think have not been fully achieved?

Yes

3. How do you find the fluid dynamics topic in general? Is it easy or difficult?

easy

4. What is the hardest thing you find about this topic?

Viscous Drag

5. If you had this program to use during class time do you think you would use it to maybe check your experimental results with the calculated results?

☑ Yes

☐ No

6. Do you think you will use this program to aid you in your revision for the exam (e.g. Use the quiz mode to test your knowledge of the topic)?

☑ Yes

☐ No

7. Are there any further improvements you wish to see in the program?

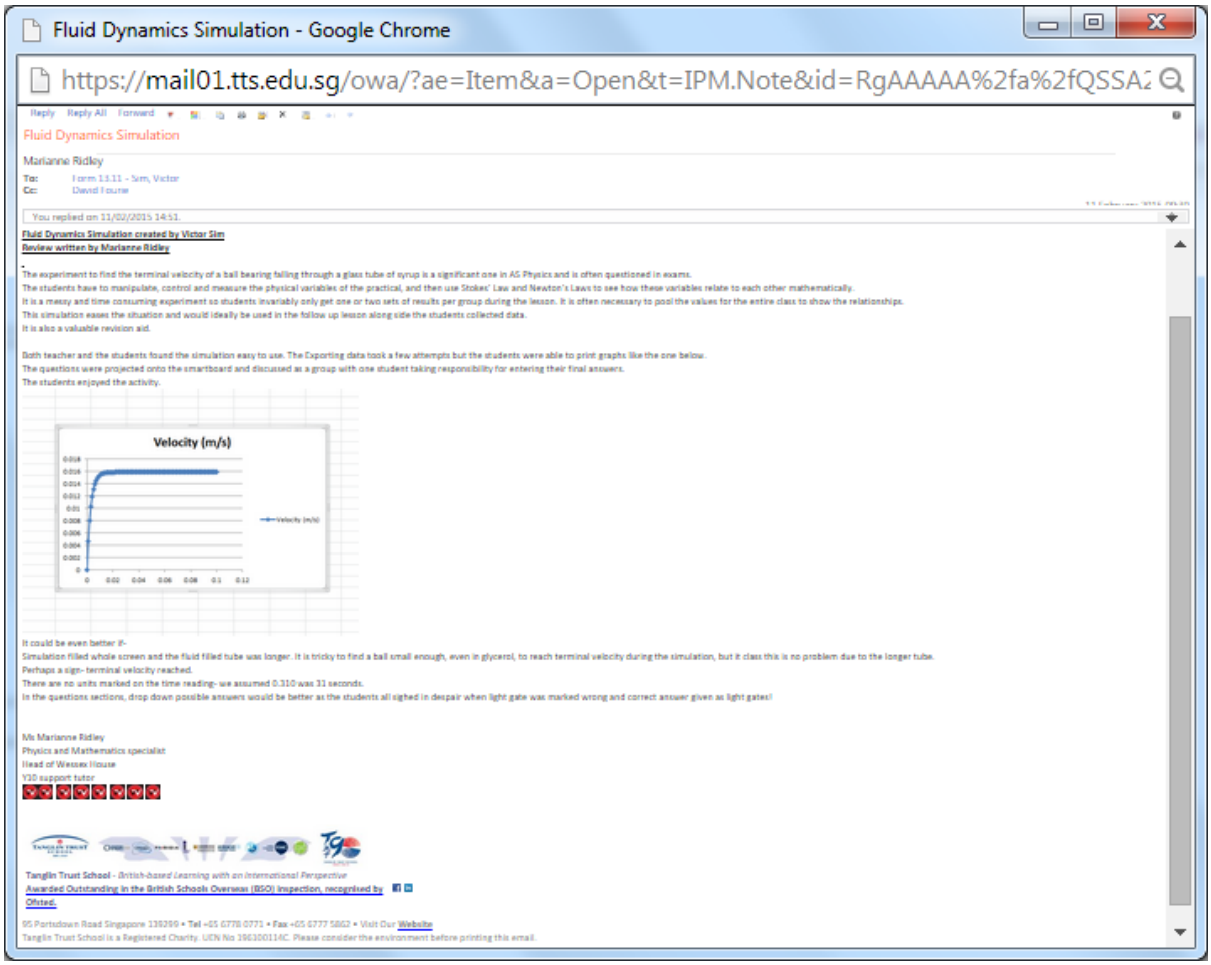
Allow multiple answers in the quiz ie a if a ~~space if~~ light gate(s) before/after

allow for data entry errors like a space after the text

→ try catch ?

☺ *Thank you for taking your time to fill out this questionnaire* ☺

7.4 E-mail



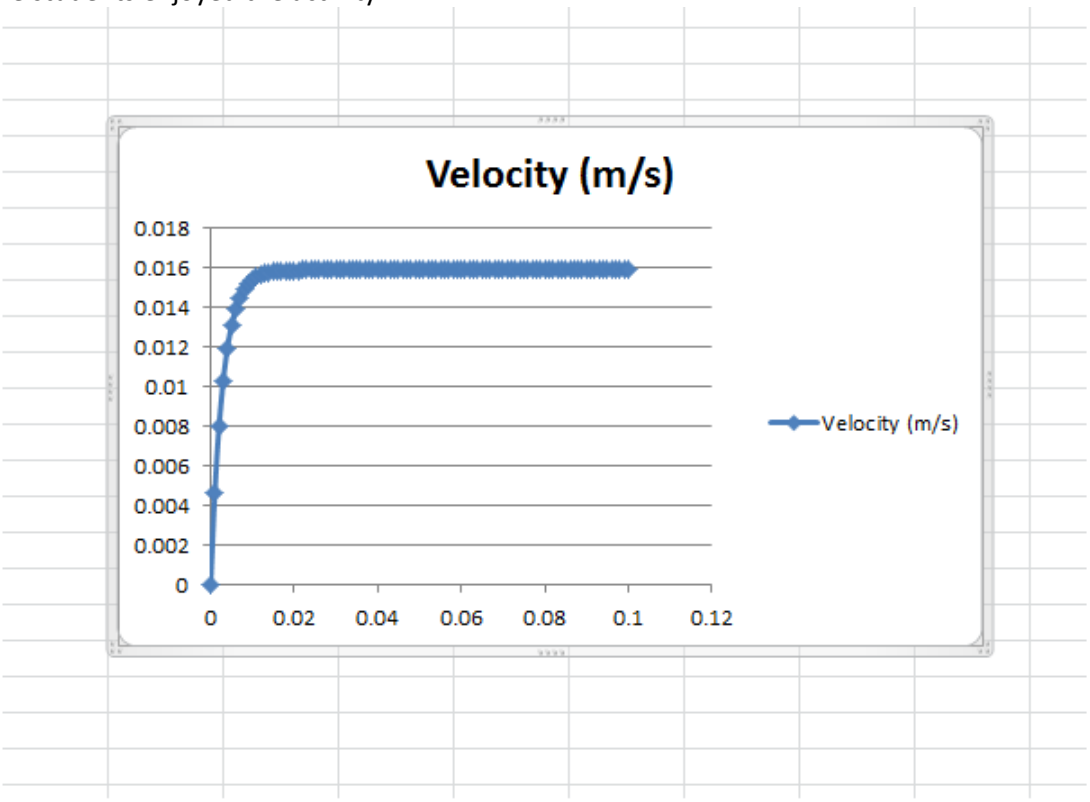
Here is the e-mail enlarged:

Fluid Dynamics Simulation created by Victor Sim
Review written by Marianne Ridley

The experiment to find the terminal velocity of a ball bearing falling through a glass tube of syrup is a significant one in AS Physics and is often questioned in exams.
The students have to manipulate, control and measure the physical variables of the practical, and then use Stokes' Law and Newton's Laws to see how these variables relate to each other mathematically.
It is a messy and time consuming experiment so students invariably only get one or two sets of results per group during the lesson. It is often necessary to pool the values for the entire class to show the relationships.
This simulation eases the situation and would ideally be used in the follow up lesson alongside the students collected data.
It is also a valuable revision aid.

Both teacher and the students found the simulation easy to use. The Exporting data took a few attempts but the students were able to print graphs like the one below.
The questions were projected onto the smartboard and discussed as a group with one student taking responsibility for entering their final answers.

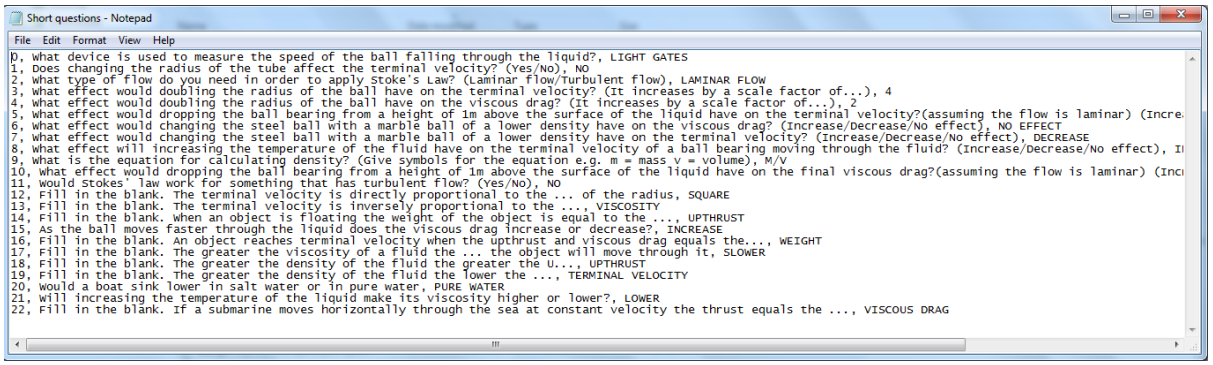
The students enjoyed the activity.



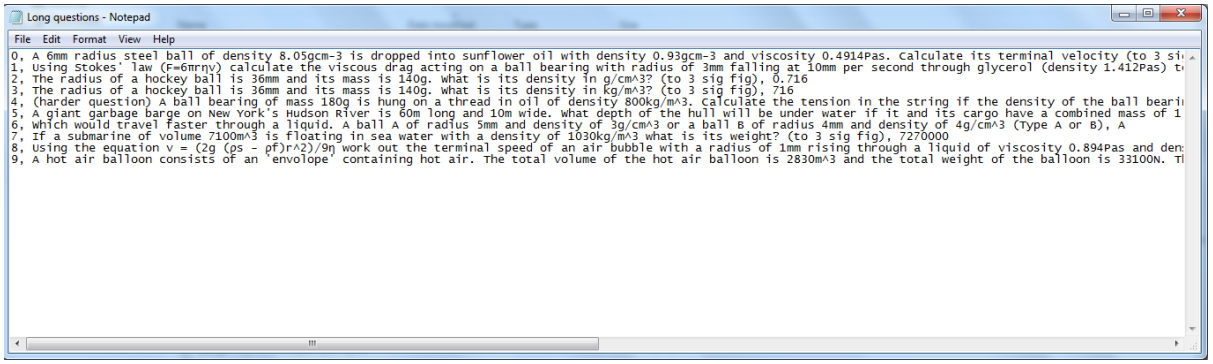
It could be even better if-
Simulation filled whole screen and the fluid filled tube was longer. It is tricky to find a ball small enough, even in glycerol, to reach terminal velocity during the simulation, but in class this is no problem due to the longer tube.
Perhaps a sign- terminal velocity reached.
There are no units marked on the time reading- we assumed 0.310 was 31 seconds.
In the questions sections, drop down possible answers would be better as the students all sighed in despair when light gate was marked wrong and correct answer given as light gates!

Ms Marianne Ridley
Physics and Mathematics specialist
Head of Wessex House
Y10 support tutor

7.5 Question files

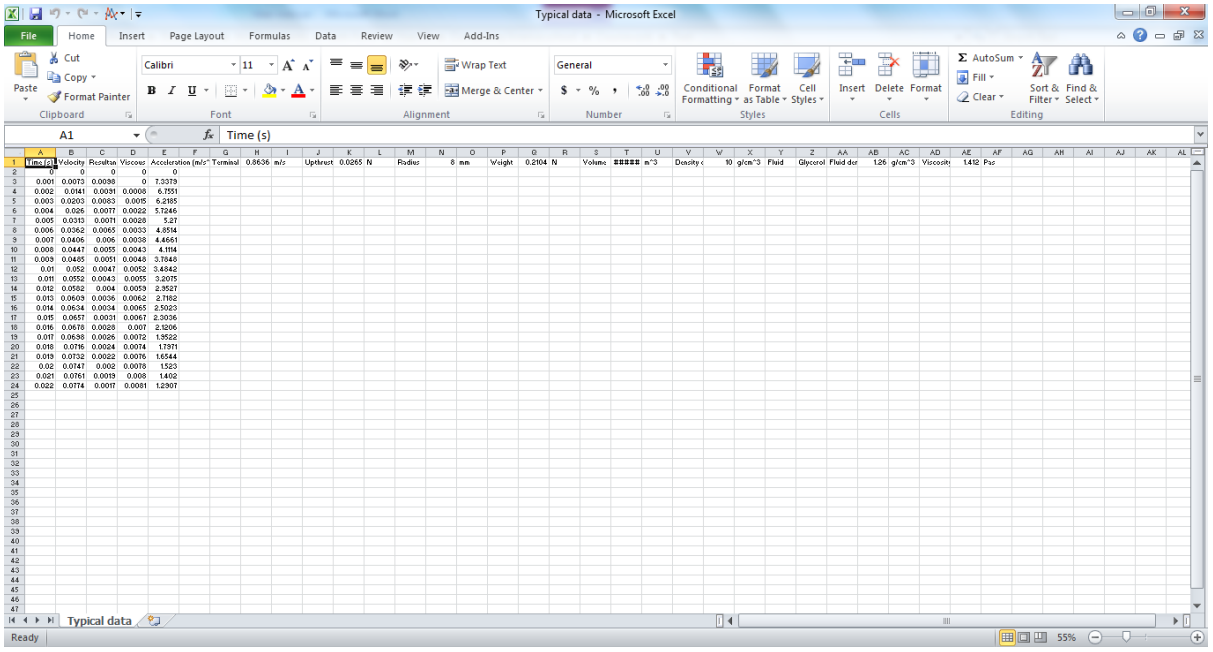


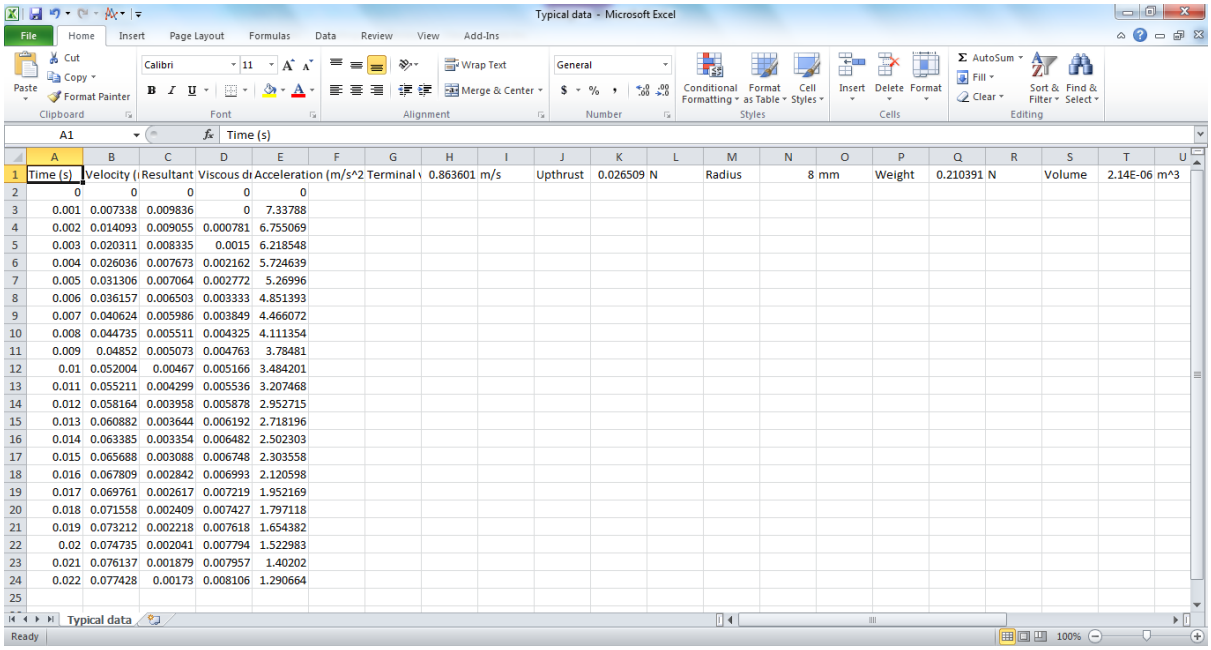
Short questions.csv file



Long questions.csv file

7.6 Exported data files





Typical data - Microsoft Excel

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	Time (s)	Velocity (r	Resultant	Viscous dr	Acceleration (m/s^2	Terminal v	0.863601	m/s		Upthrust	0.026509	N	Radius	8	mm	Weight	0.210391	N	Volume	2.14E-06	m^3
2	0	0	0	0	0																
3	0.001	0.007338	0.009836	0	7.33788																
4	0.002	0.014093	0.009055	0.000781	6.755069																
5	0.003	0.020311	0.008335	0.0015	6.218548																
6	0.004	0.026036	0.007673	0.002162	5.724639																
7	0.005	0.031306	0.007064	0.002772	5.26996																
8	0.006	0.036157	0.006503	0.003333	4.851393																
9	0.007	0.040624	0.005986	0.003849	4.466072																
10	0.008	0.044735	0.005511	0.004325	4.111354																
11	0.009	0.04852	0.005073	0.004763	3.78481																
12	0.01	0.052004	0.00467	0.005166	3.484201																
13	0.011	0.055211	0.004299	0.005536	3.207468																
14	0.012	0.058164	0.003958	0.005878	2.952715																
15	0.013	0.060882	0.003644	0.006192	2.718196																
16	0.014	0.063385	0.003354	0.006482	2.502303																
17	0.015	0.065688	0.003088	0.006748	2.303558																
18	0.016	0.067809	0.002842	0.006993	2.120598																
19	0.017	0.069761	0.002617	0.007219	1.952169																
20	0.018	0.071558	0.002409	0.007427	1.797118																
21	0.019	0.073212	0.002218	0.007618	1.654382																
22	0.02	0.074735	0.002041	0.007794	1.522983																
23	0.021	0.076137	0.001879	0.007957	1.40202																
24	0.022	0.077428	0.00173	0.008106	1.290664																
25																					

Typical data

Ready

A zoomed in version of the previous picture

7.7 Testing appendix

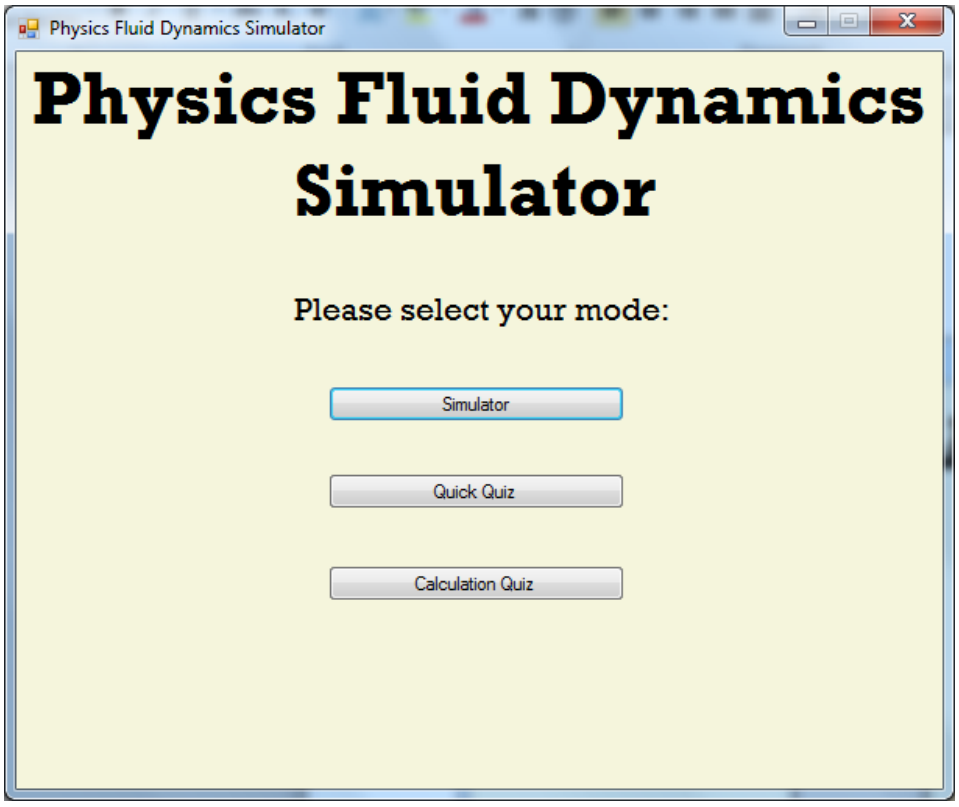


Fig 1.0 (screenshot of main menu)

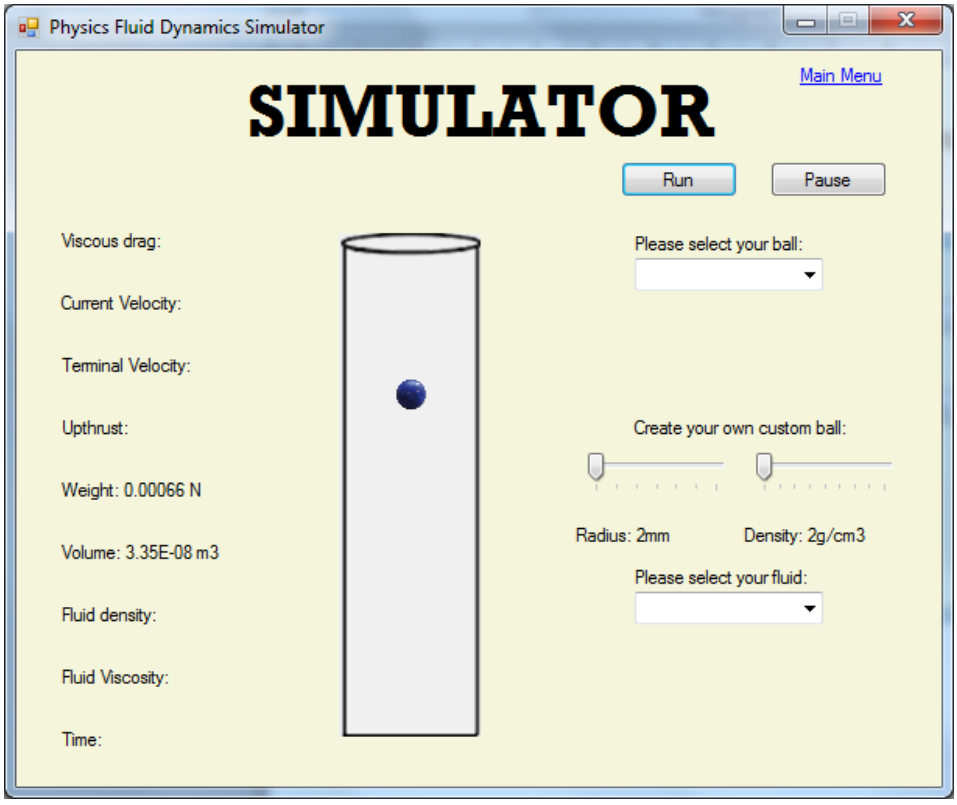


Fig 1.1 (showing it goes from main menu to simulator mode)

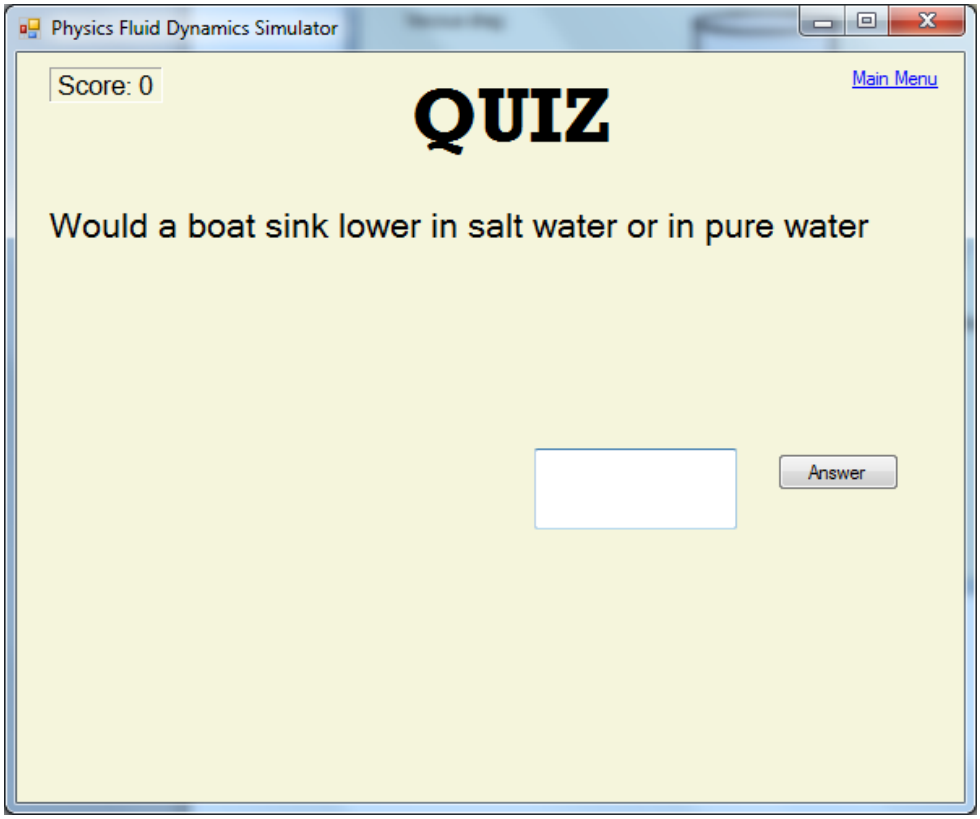


Fig 1.2 (showing it goes from main menu to quick quiz mode)

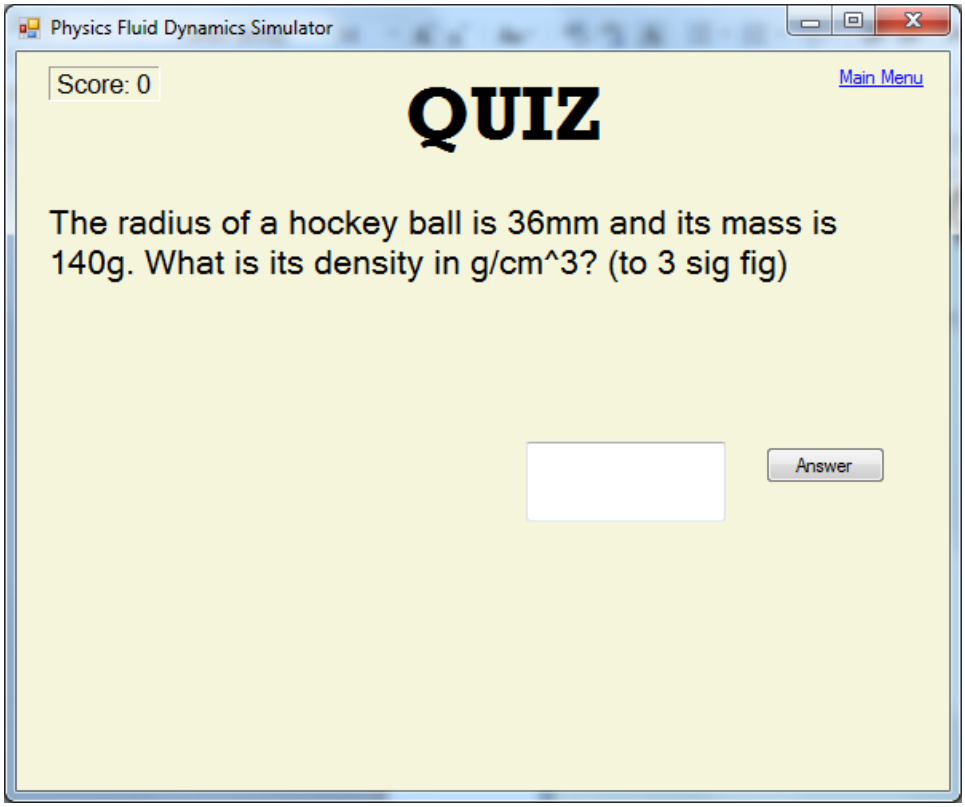


Fig 1.3 (showing it goes from main menu to calculation quiz mode)

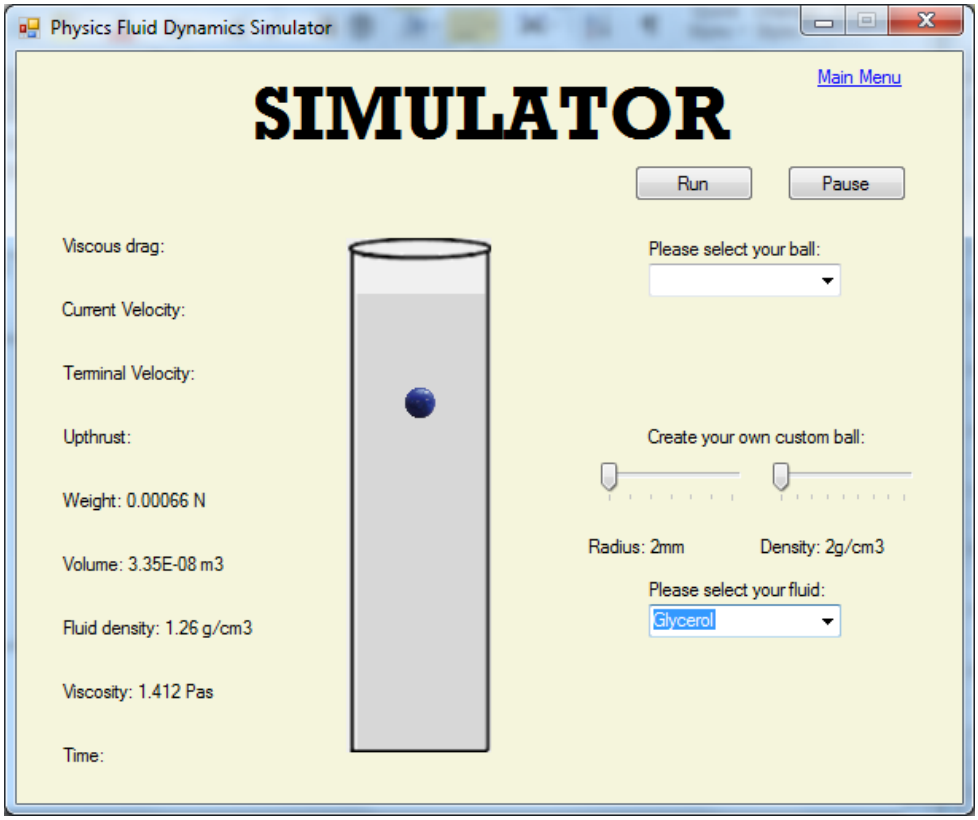


Fig 2.10

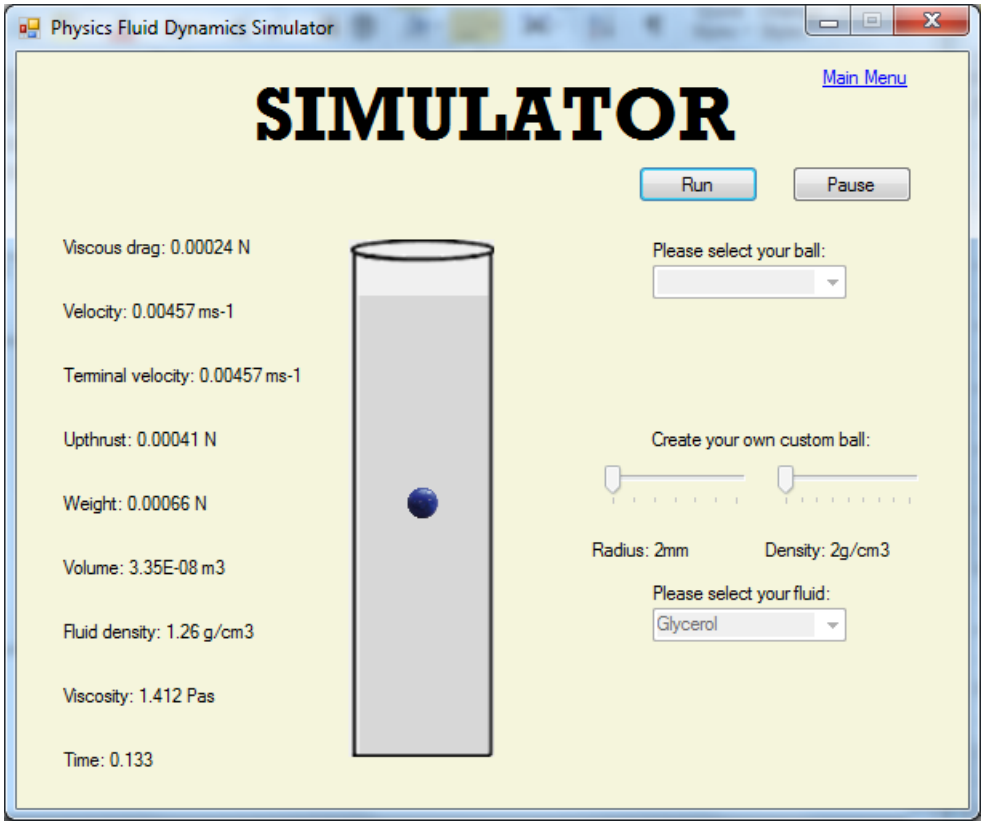


Fig 2.11 (sinks in glycerol)

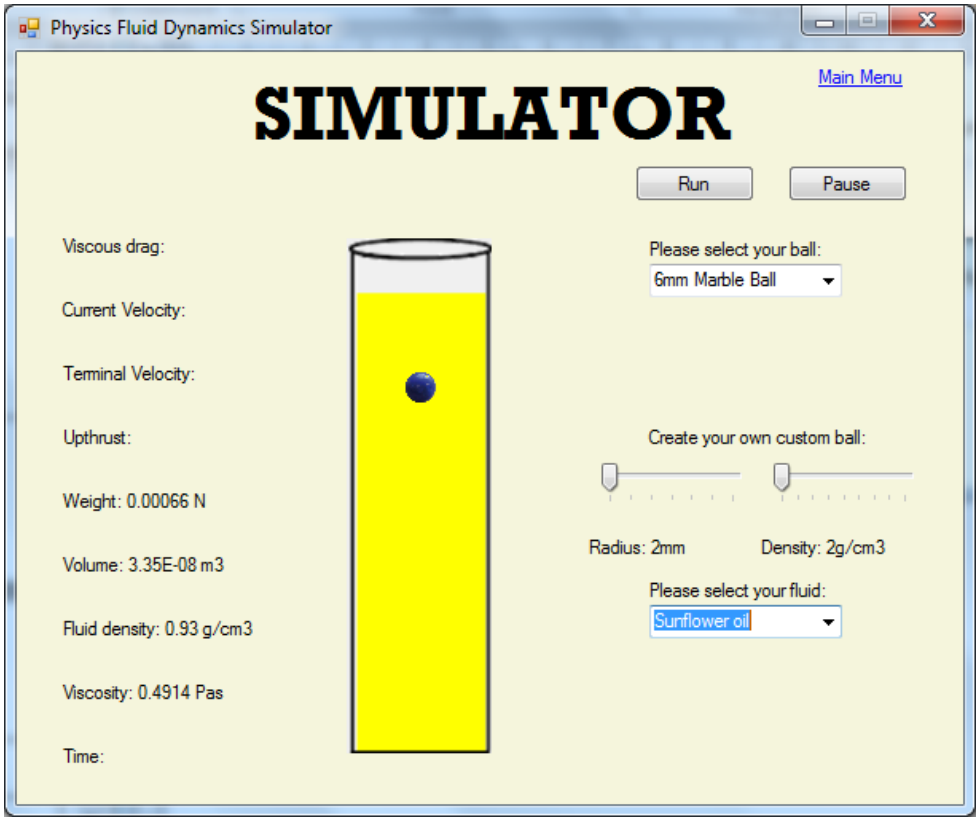


Fig 2.12

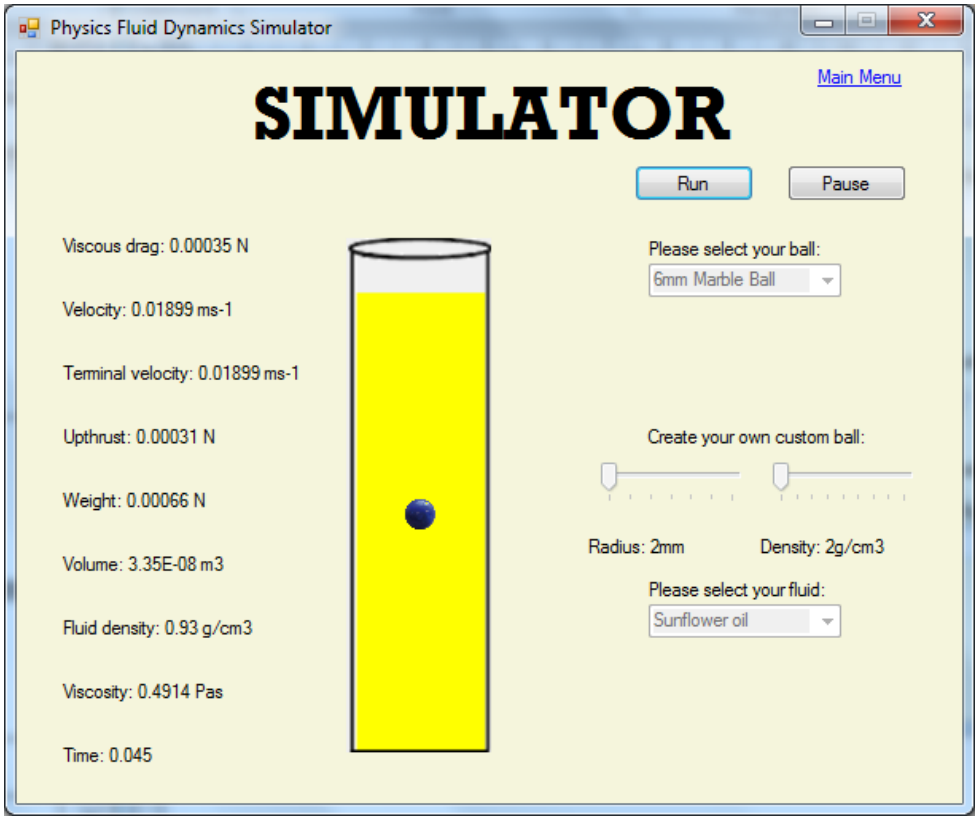


Fig 2.13 (sinks in sunflower oil)

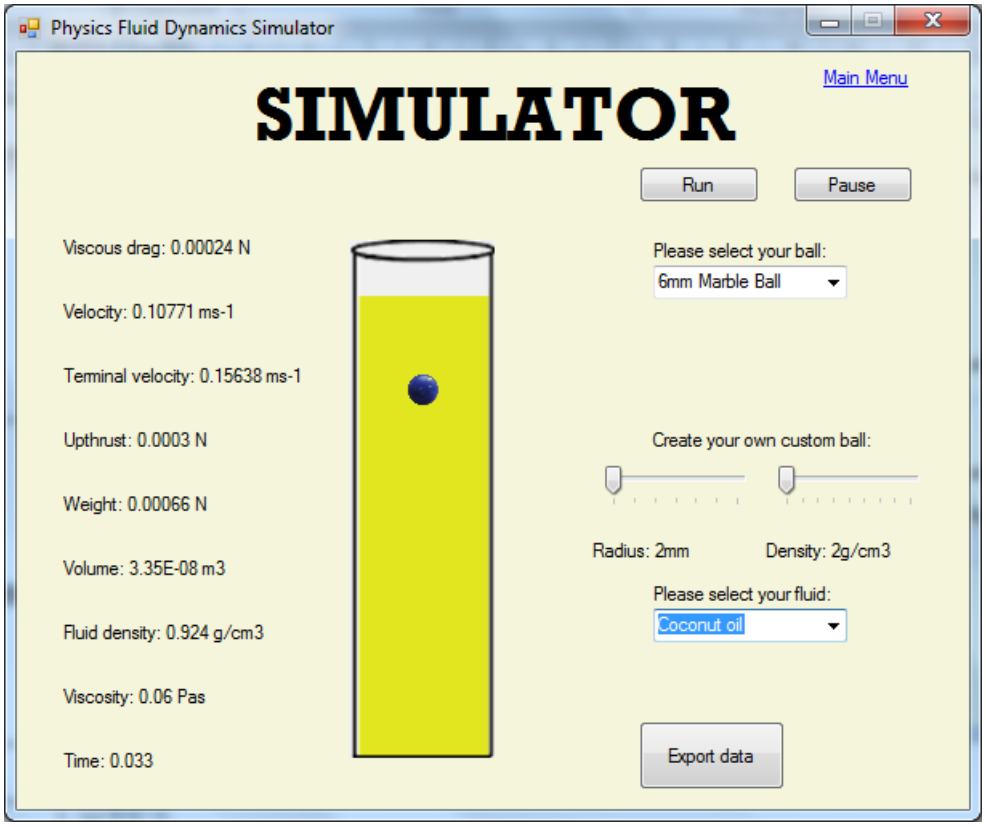


Fig 2.14

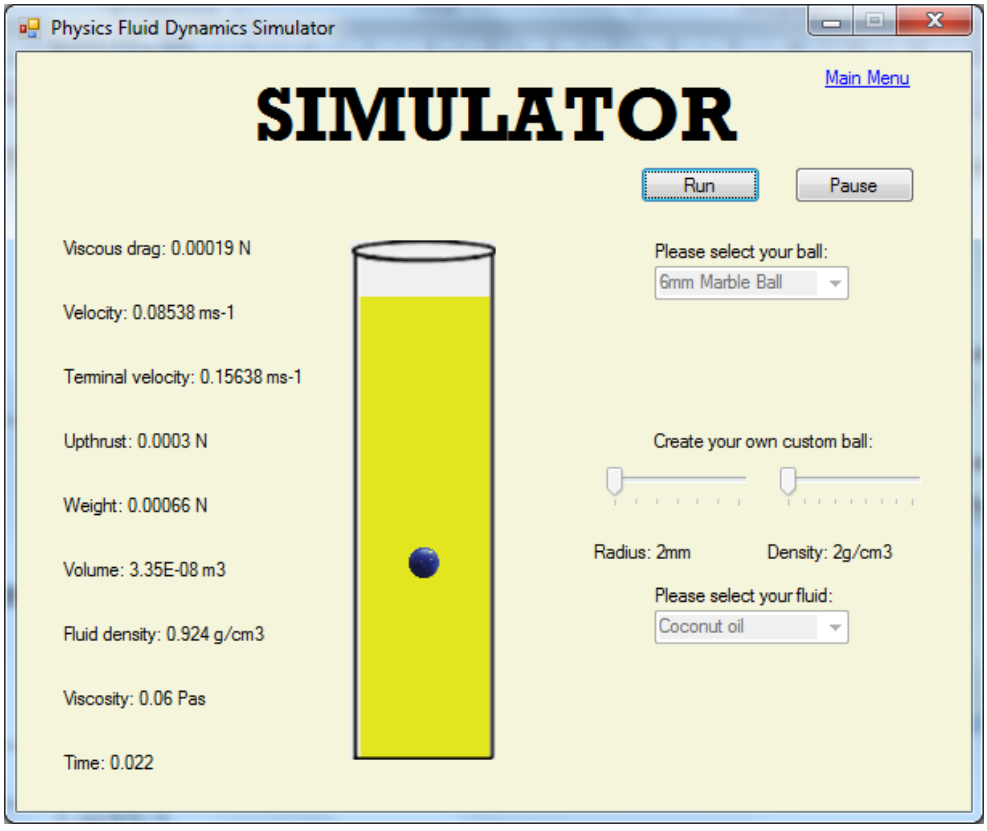


Fig 2.15 (sinks in coconut oil)

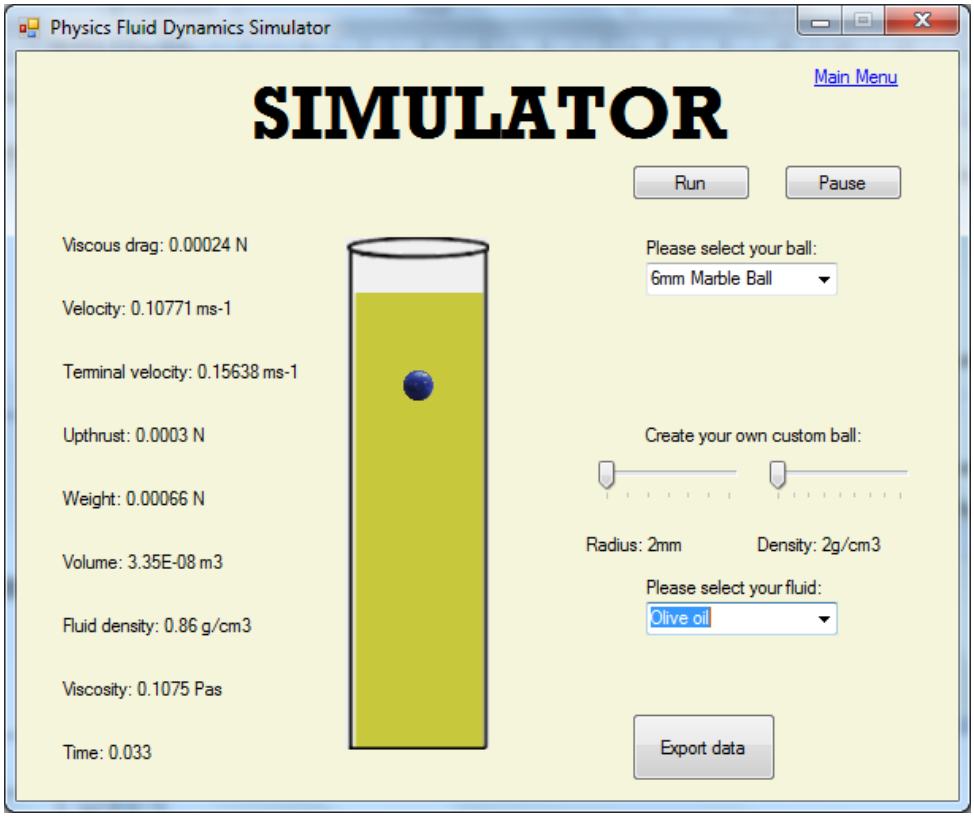


Fig 2.16

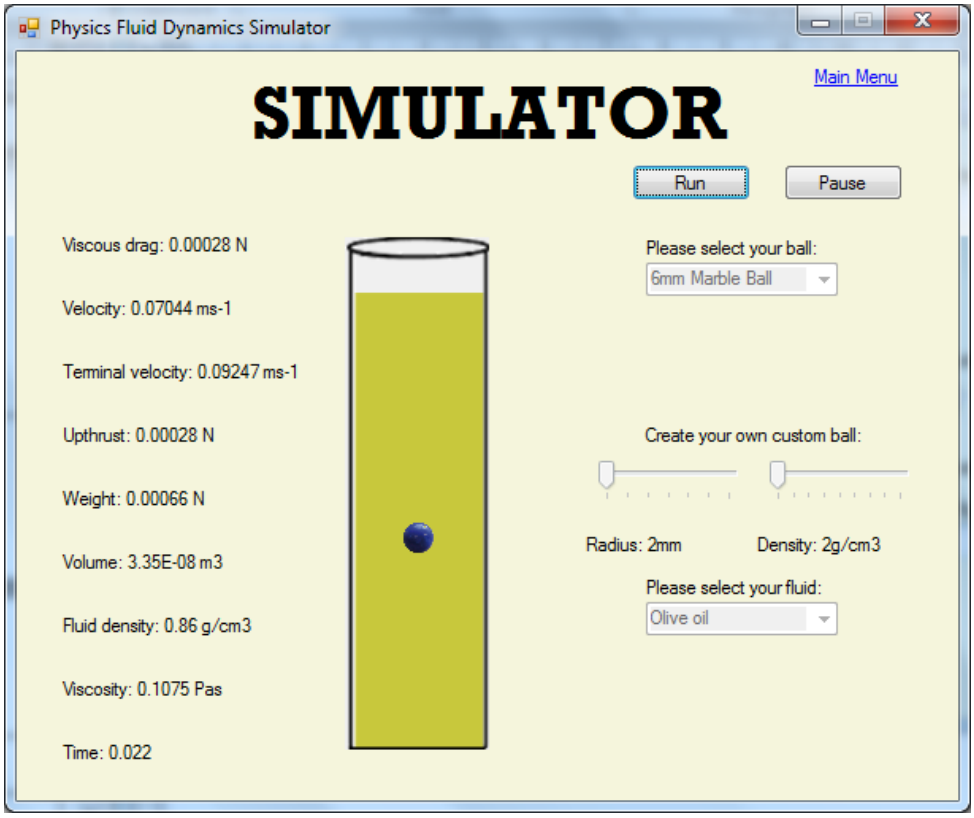


Fig 2.17 (sinks in olive oil)

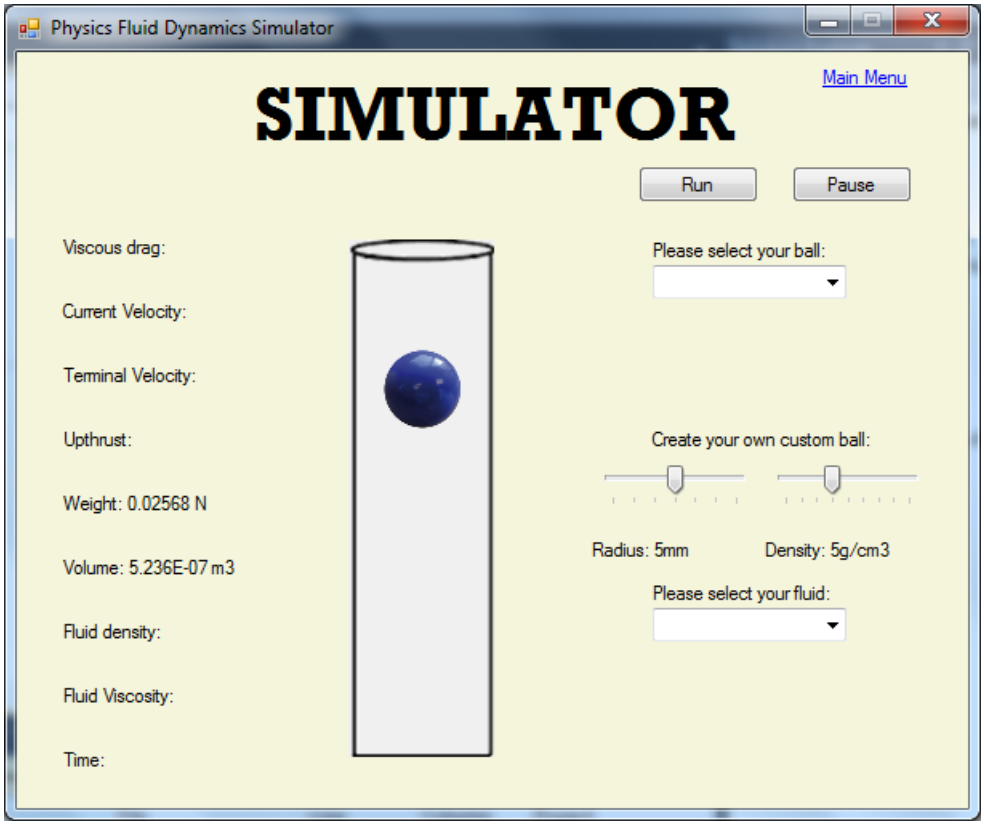


Fig 2.18 (correct weight and volume produced)

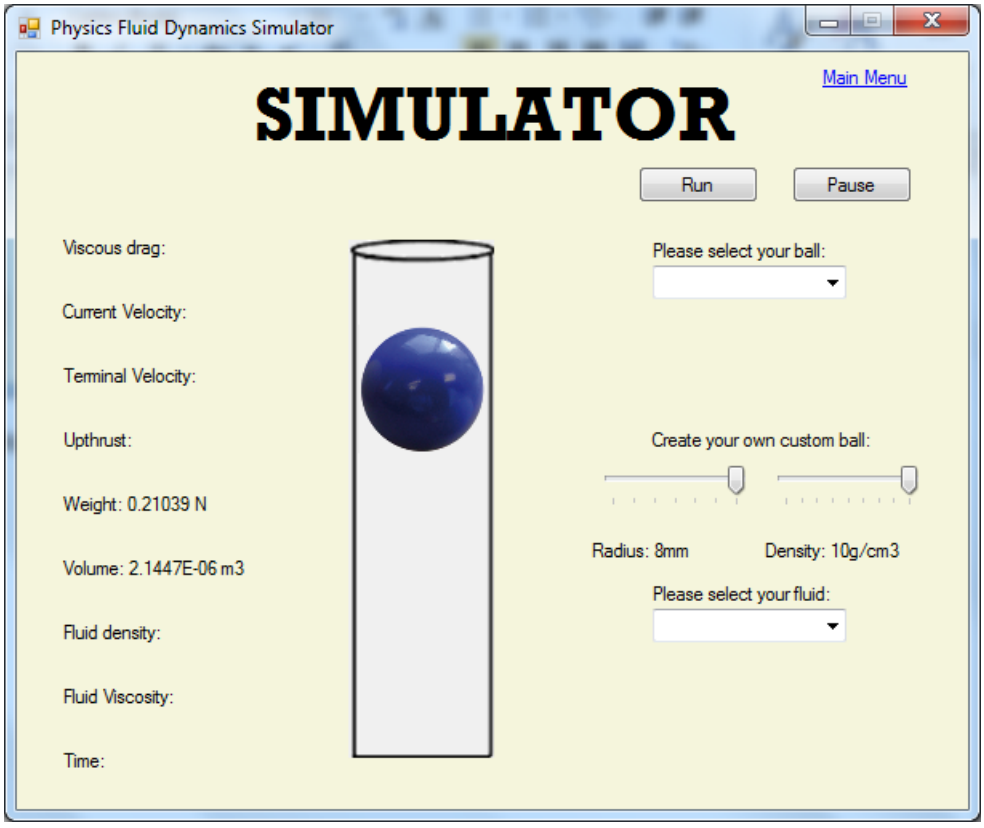


Fig 2.19 (correct weight and volume produced)

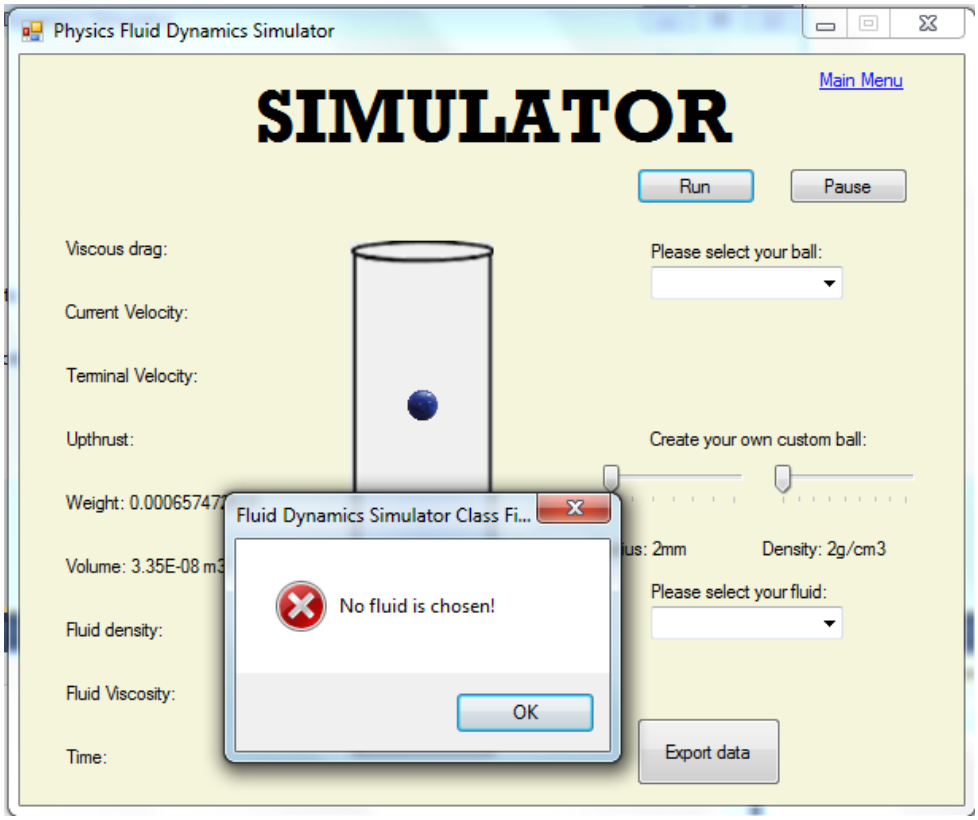


Fig 2.20 (appropriate error message displayed)

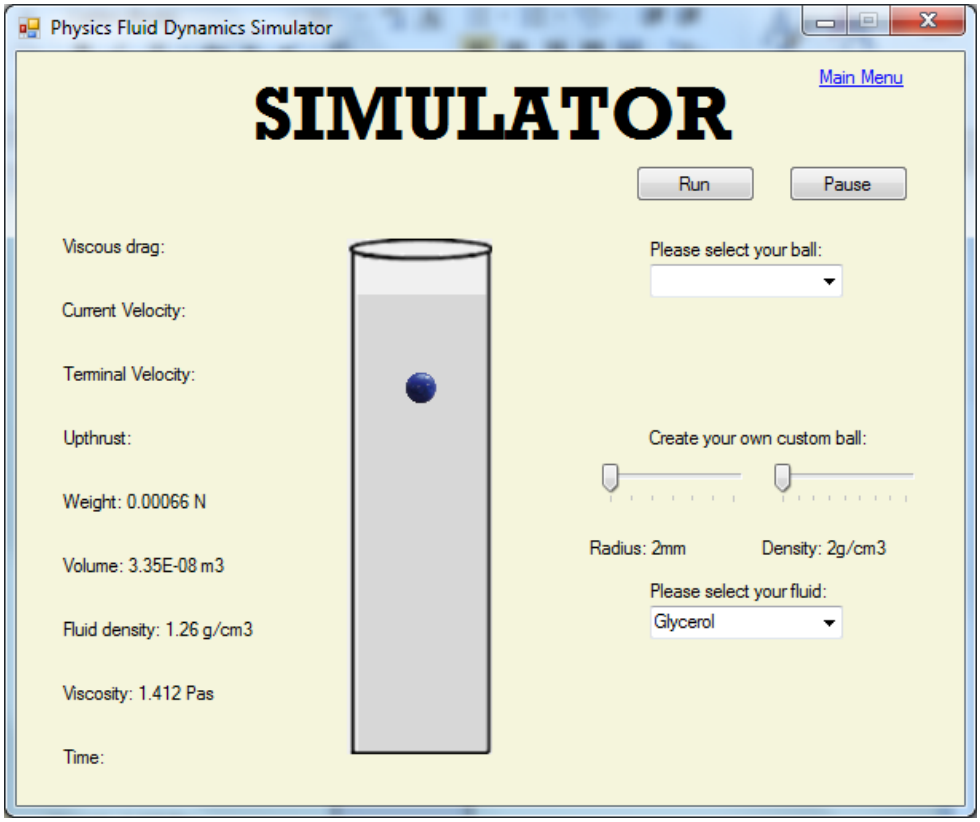


Fig 2.21 (Fluid density, viscosity and colour of fluid all changed accordingly)

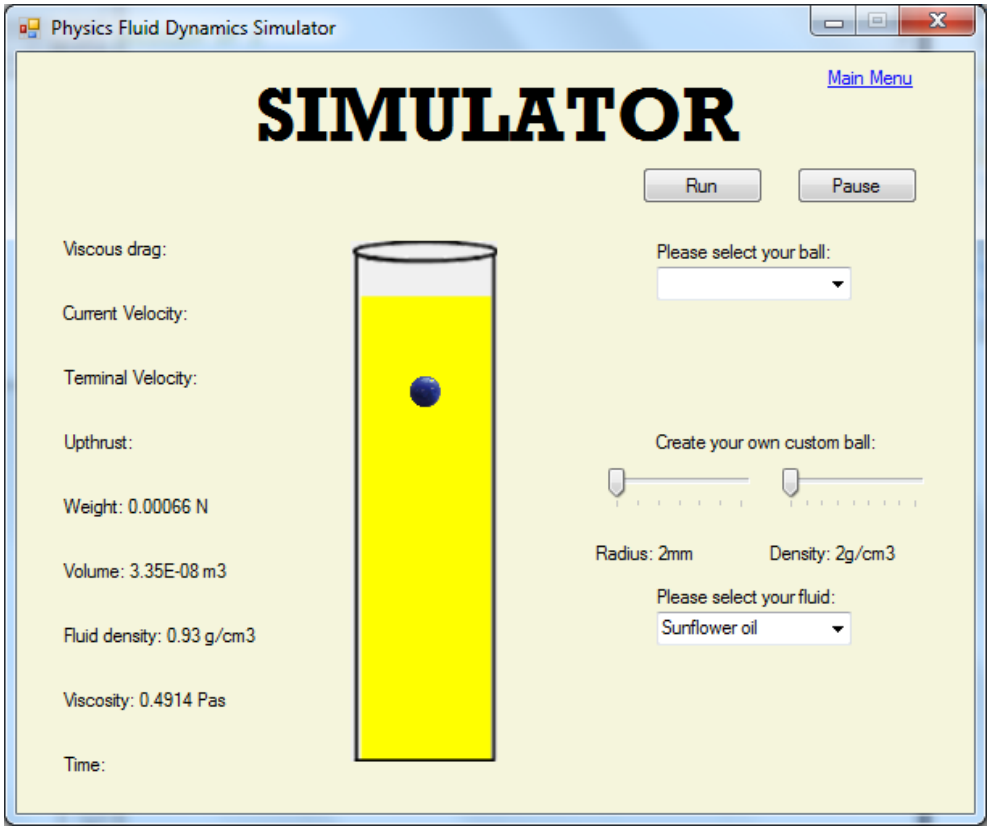


Fig 2.22 (Fluid density, viscosity and colour of fluid all changed accordingly)

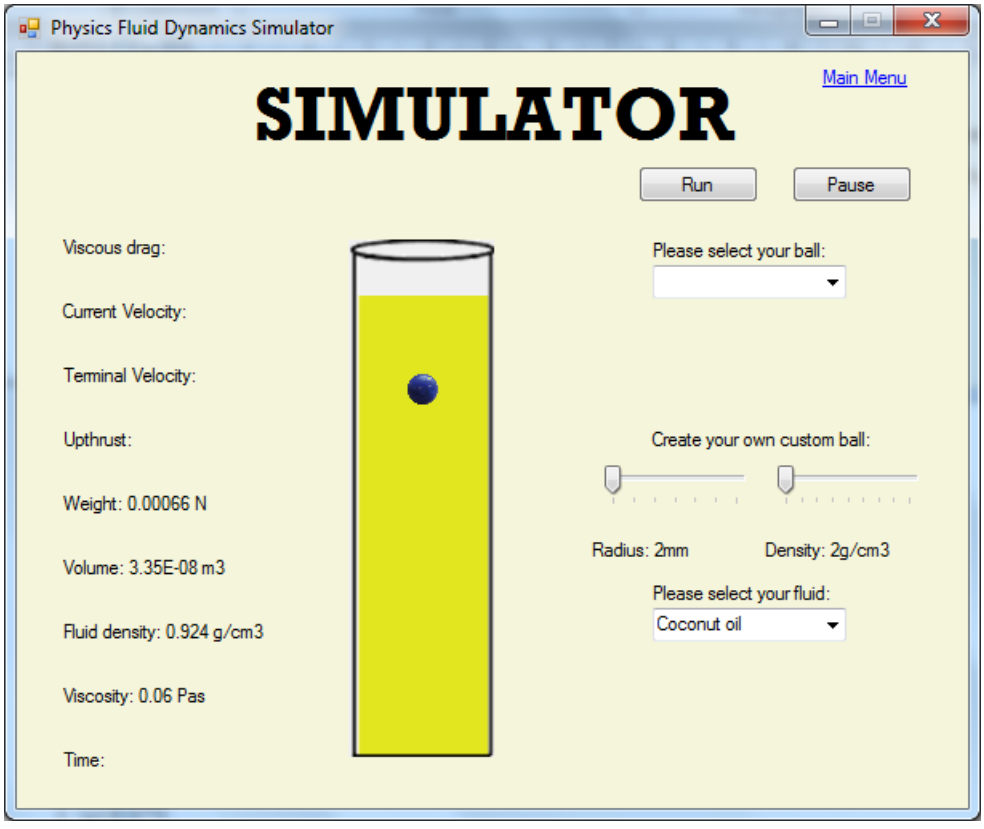


Fig 2.23 (Fluid density, viscosity and colour of fluid all changed accordingly)

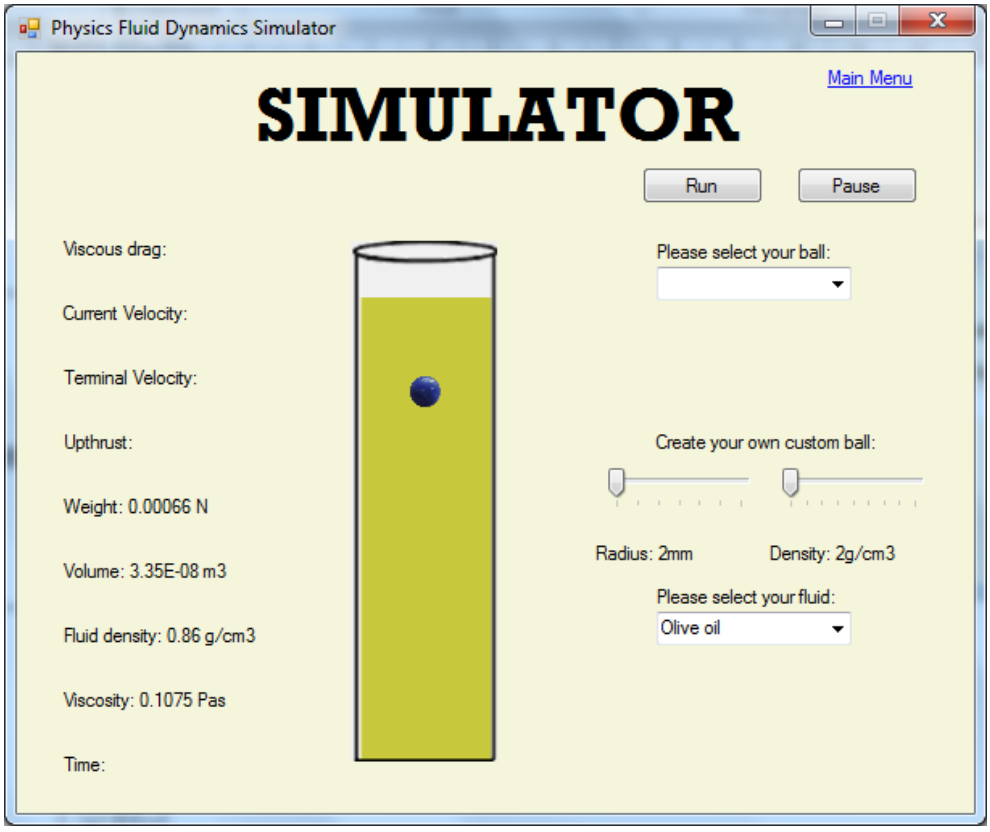


Fig 2.24 (Fluid density, viscosity and colour of fluid all changed accordingly)

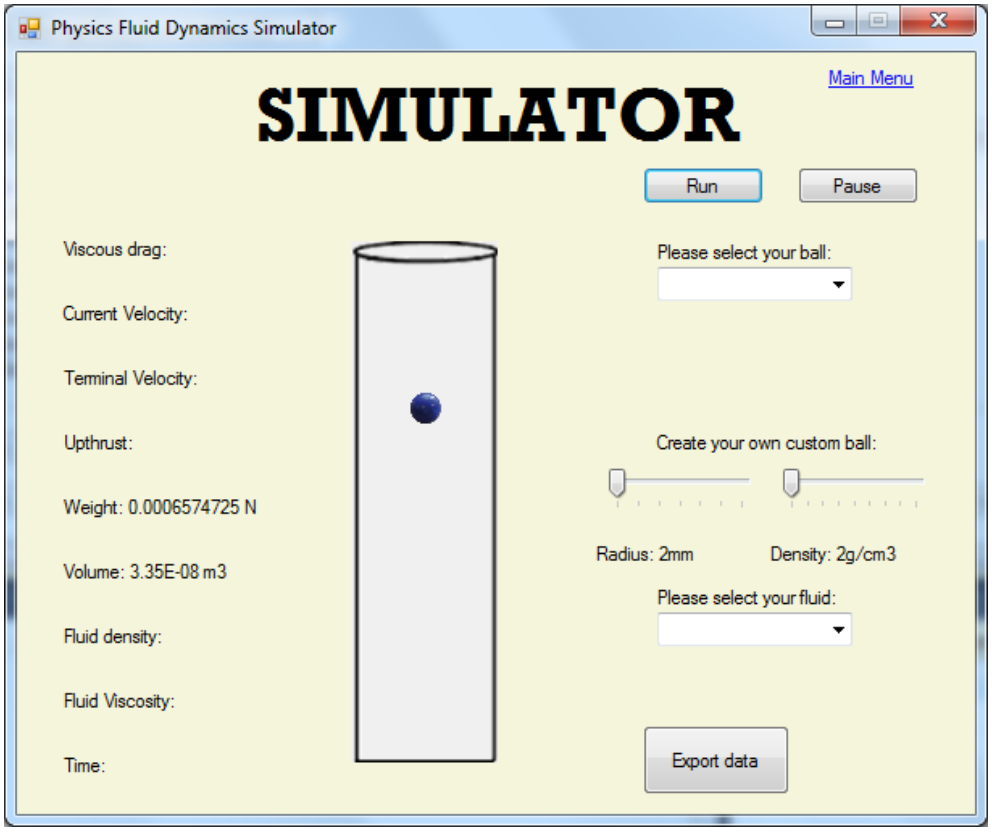


Fig 2.25 (screenshot showing original size of ball)

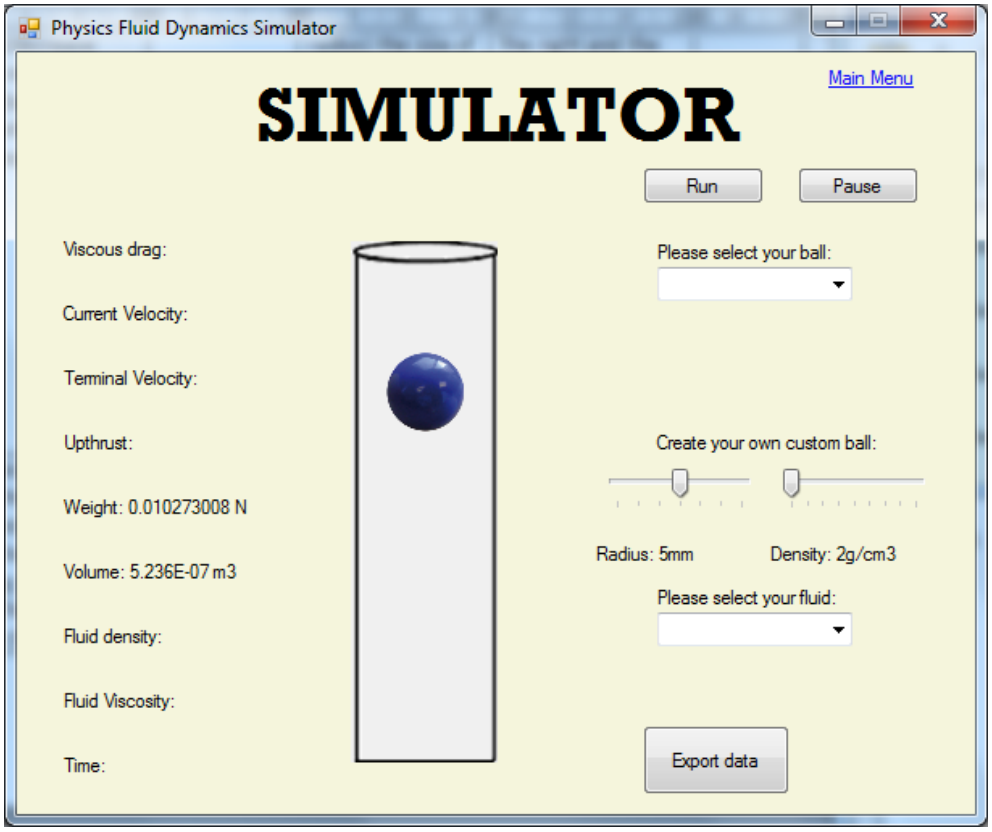


Fig 2.26 (screenshot showing size of ball after radius has been adjusted)

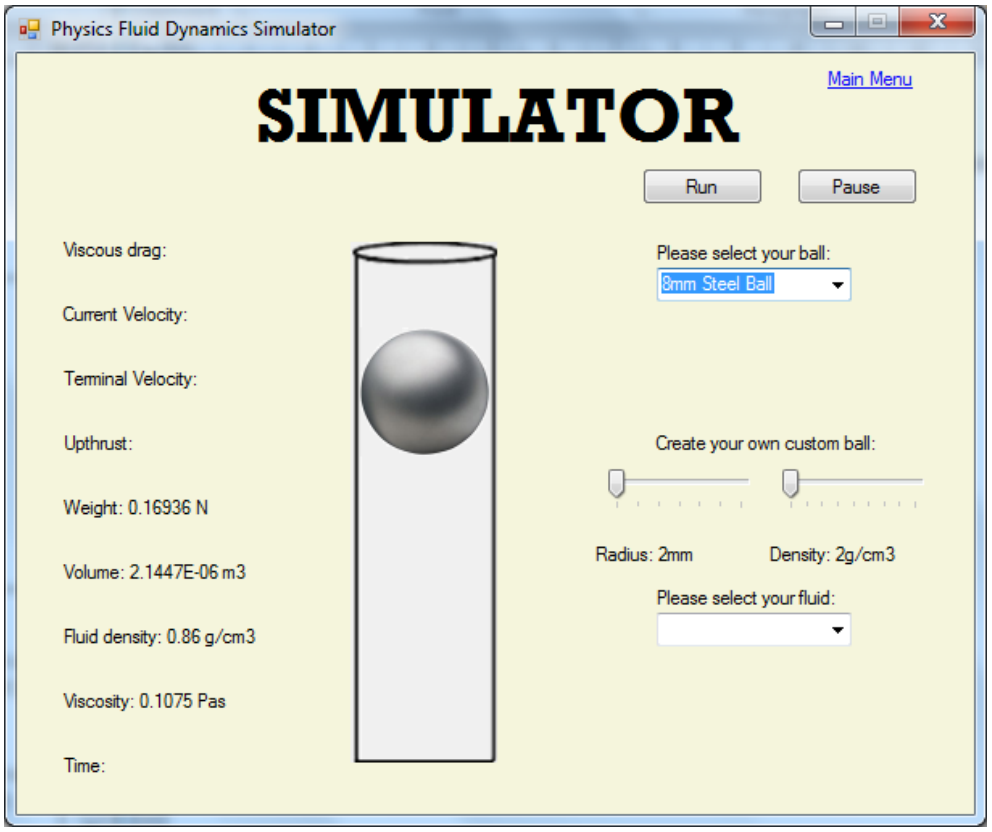


Fig 2.27 (screenshot of it showing that the ball has changed to a steel ball when steel ball is selected)

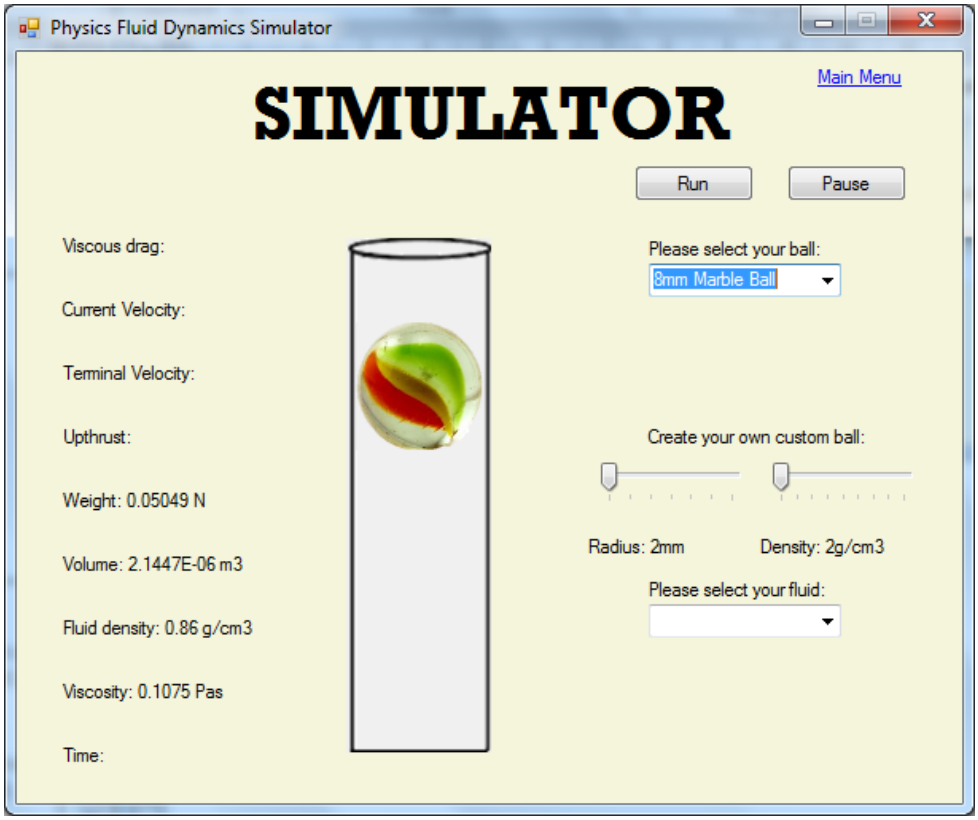


Fig 2.28 (screenshot showing that it has changed into a marble ball when marble ball is selected)

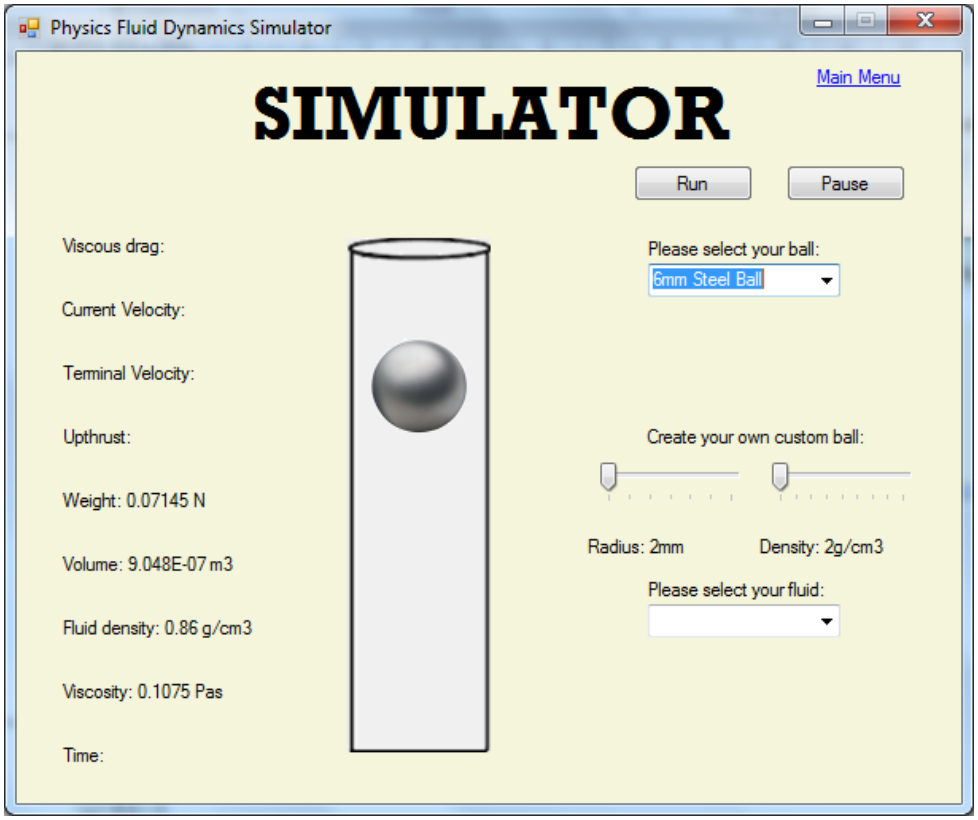


Fig 2.29 (screenshot of ball becoming steel ball also with 6mm radius)

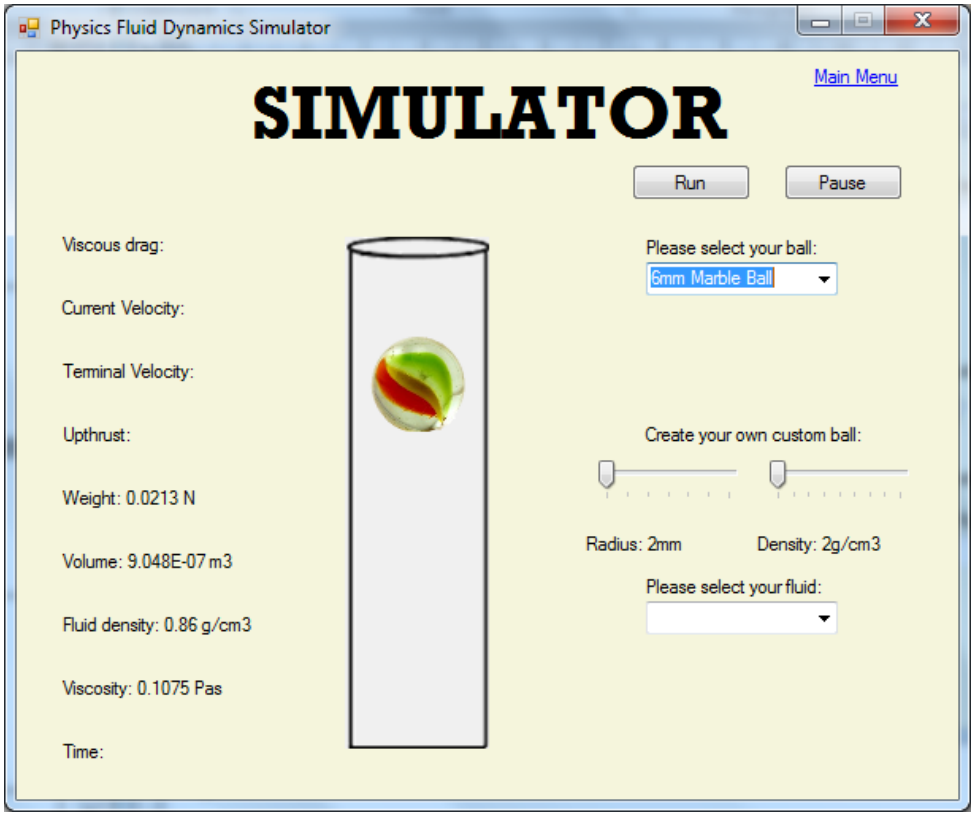


Fig 2.30 (screenshot of the ball becoming marble ball also with 6mm radius)

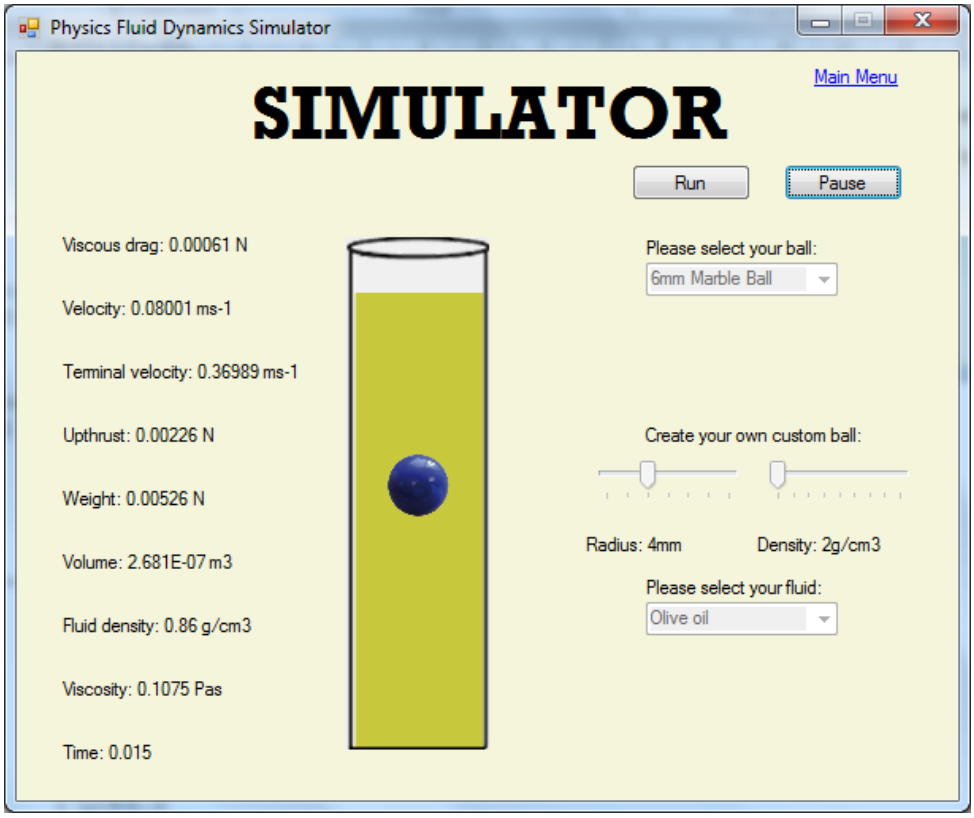


Fig 2.31 (showing the simulation paused)

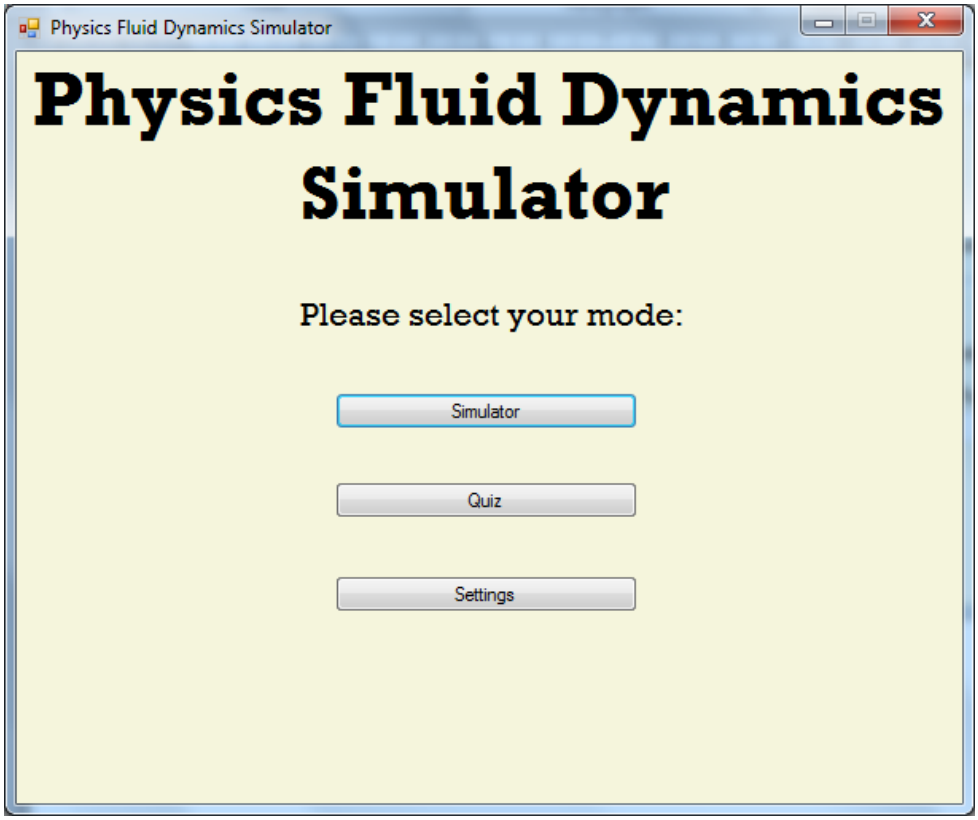


Fig 2.32 (starts from main menu)

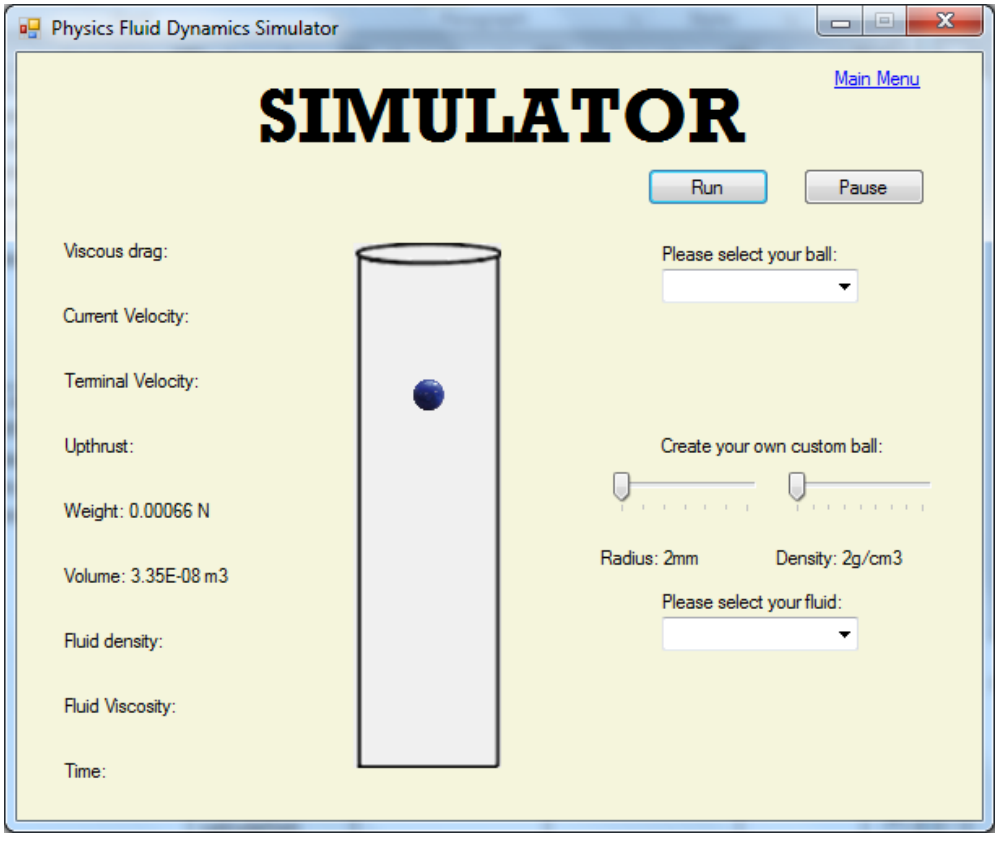


Fig 2.33 (goes to simulator mode)

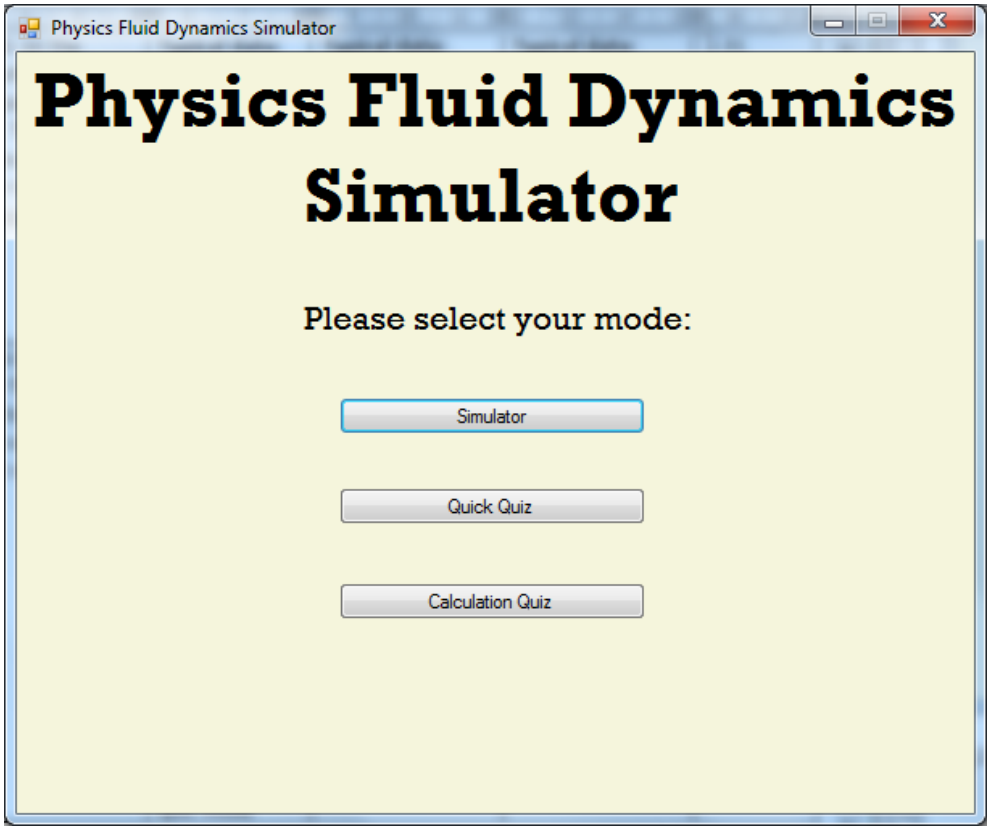
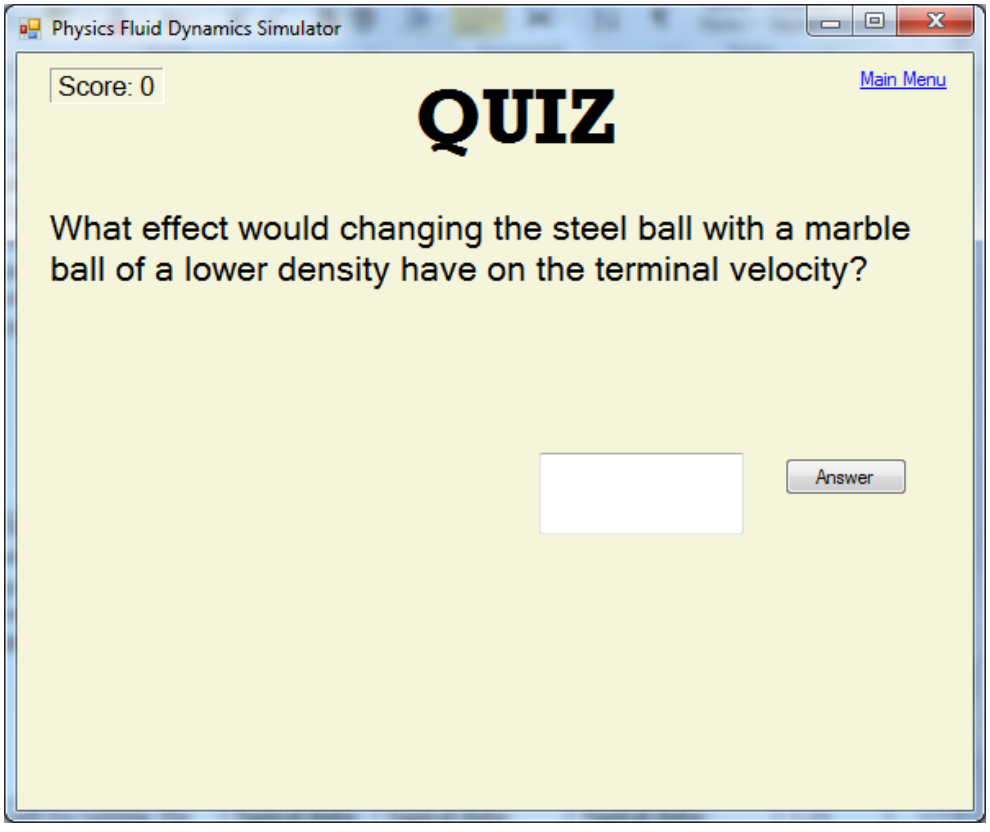


Fig 2.34 (gets back to main menu by clicking on the main menu link on simulator mode)



2.35 (goes to quick quiz mode)

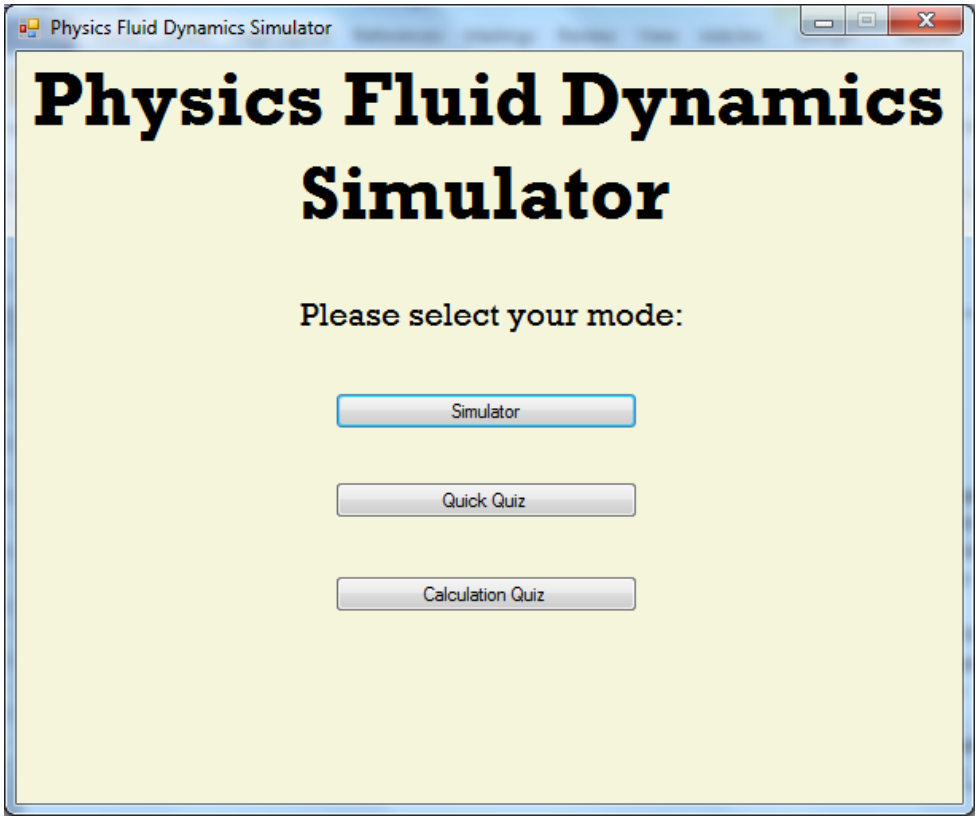


Fig 2.36 (gets back to main menu by clicking the main menu link on quick quiz mode)

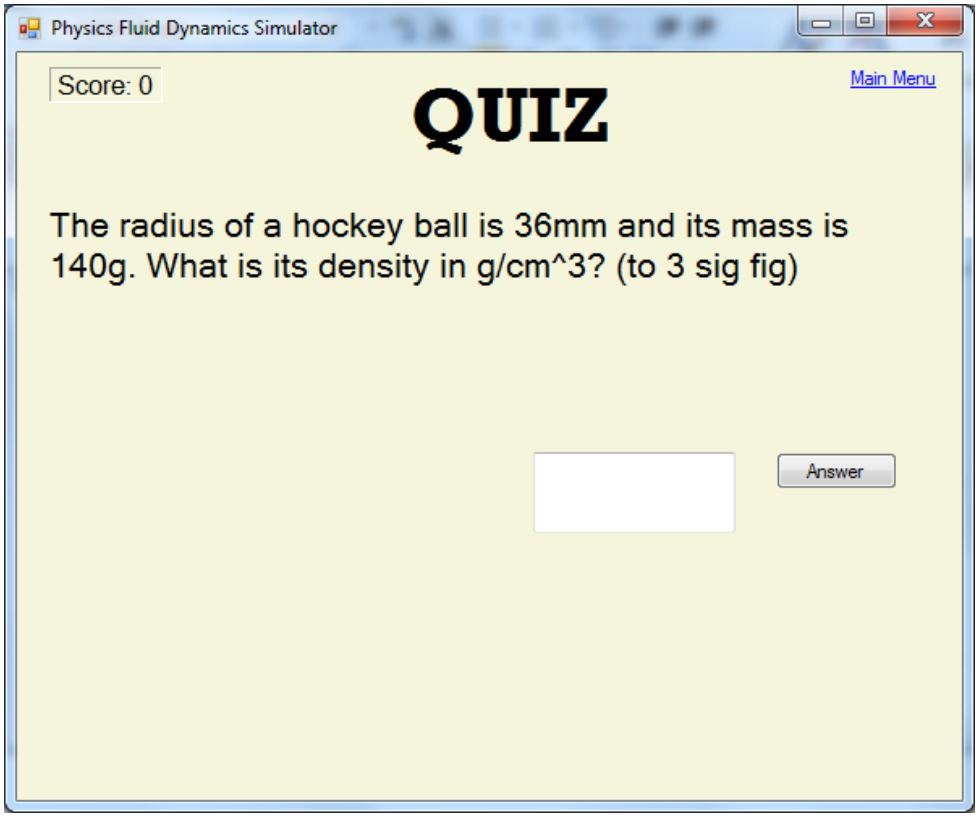


Fig 2.37 (goes to calculation quiz mode)

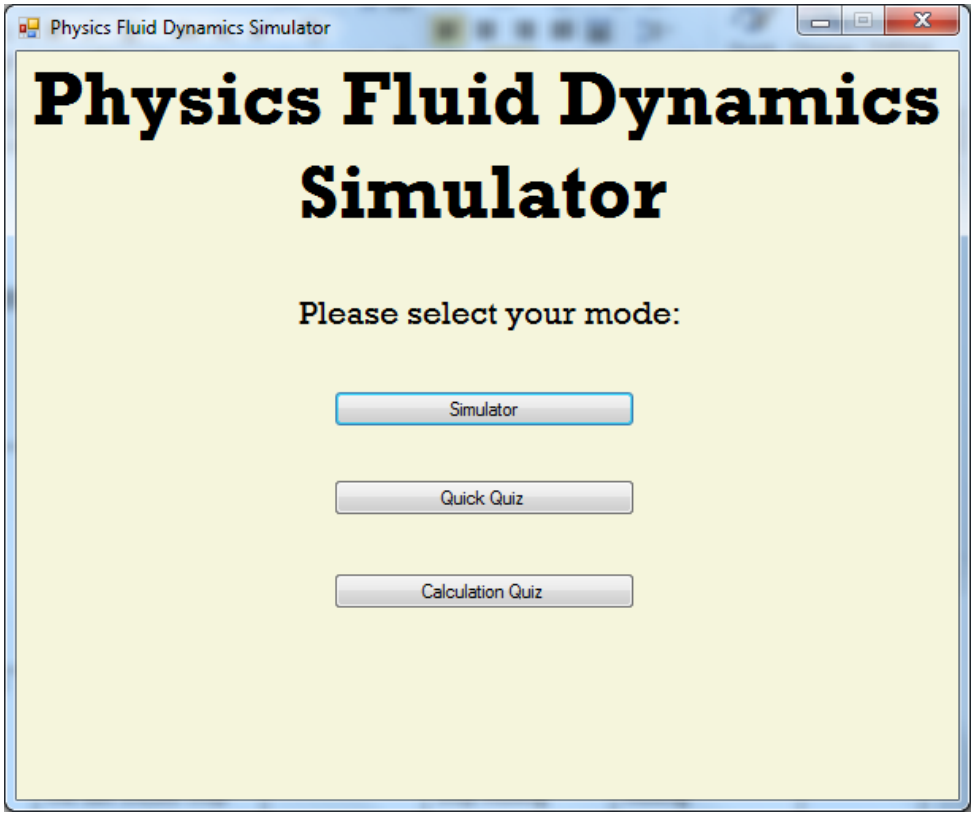


Fig 2.38 (goes back to main menu by clicking on the main menu link on calculation quiz mode)

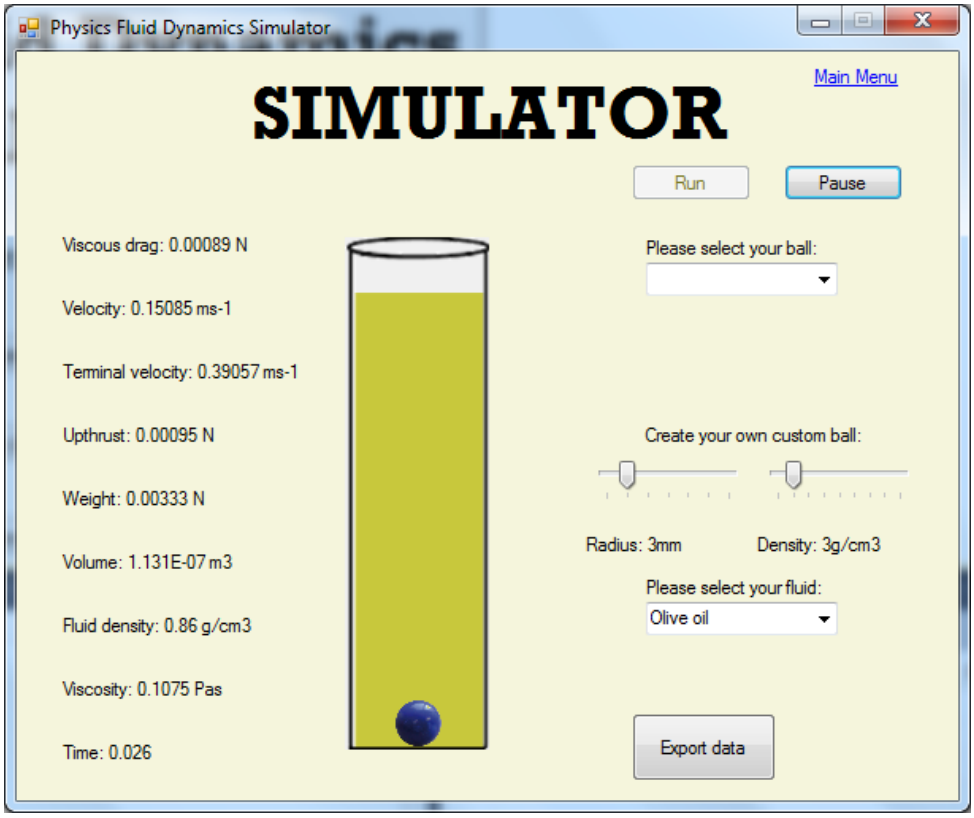


Fig 2.39 (The ball stopped at the bottom of the tube without going pass it and the simulation ended)

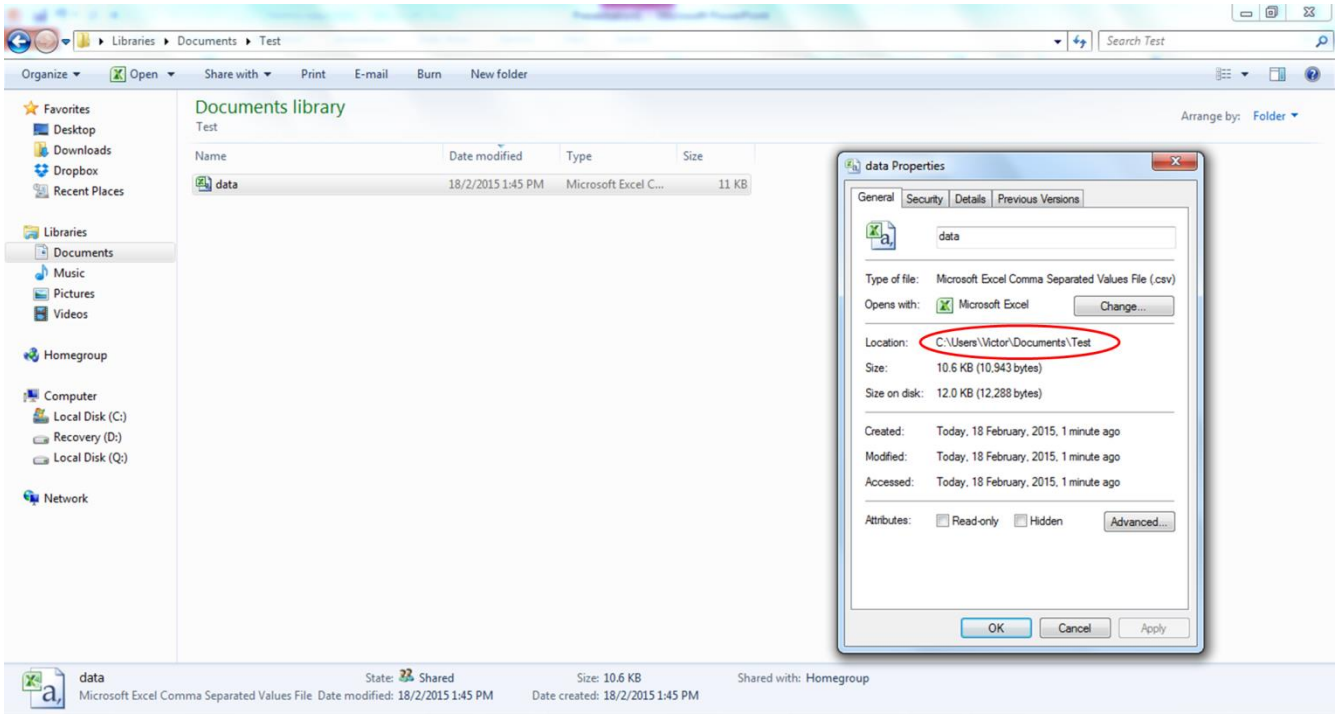


Fig 2.40 (shows a csv file created which holds the data from the simulation)

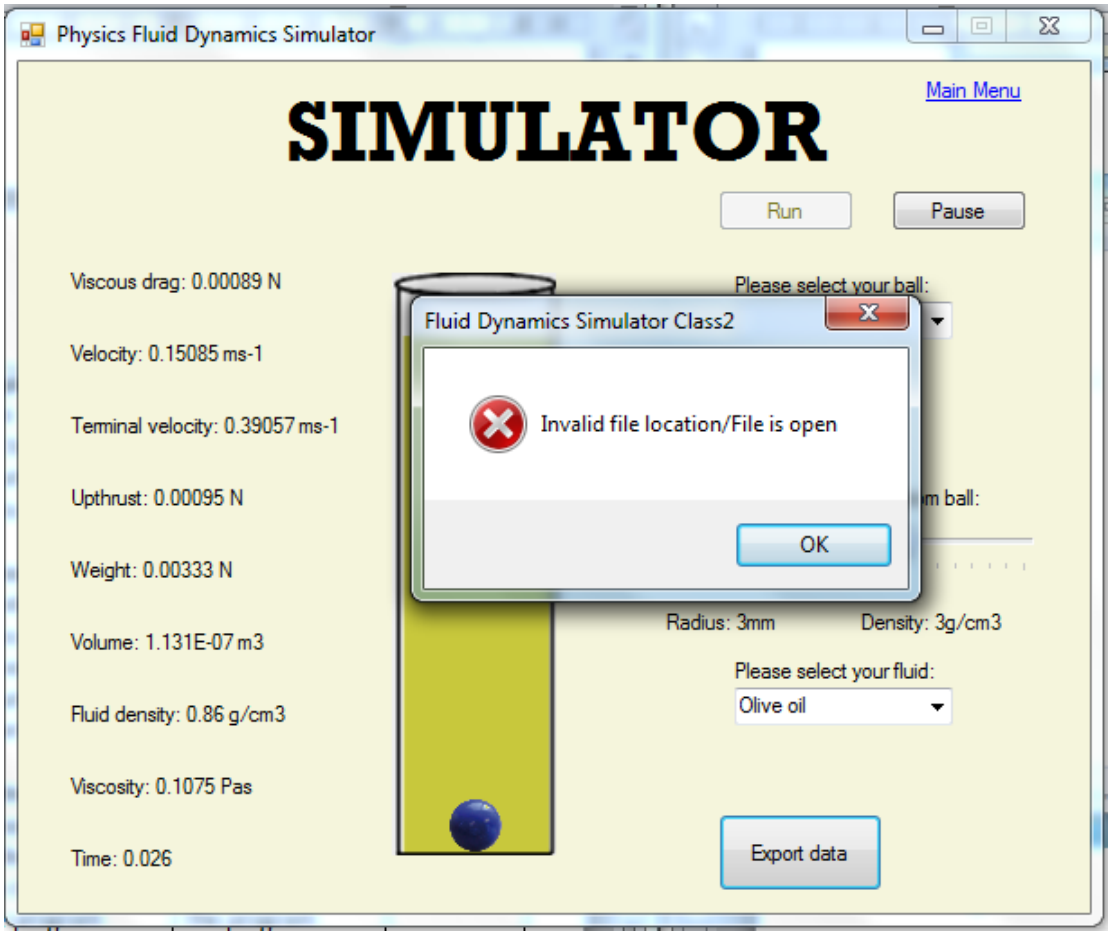


Fig 2.41 (error message showing up telling the user the file he wants to write to is currently open)

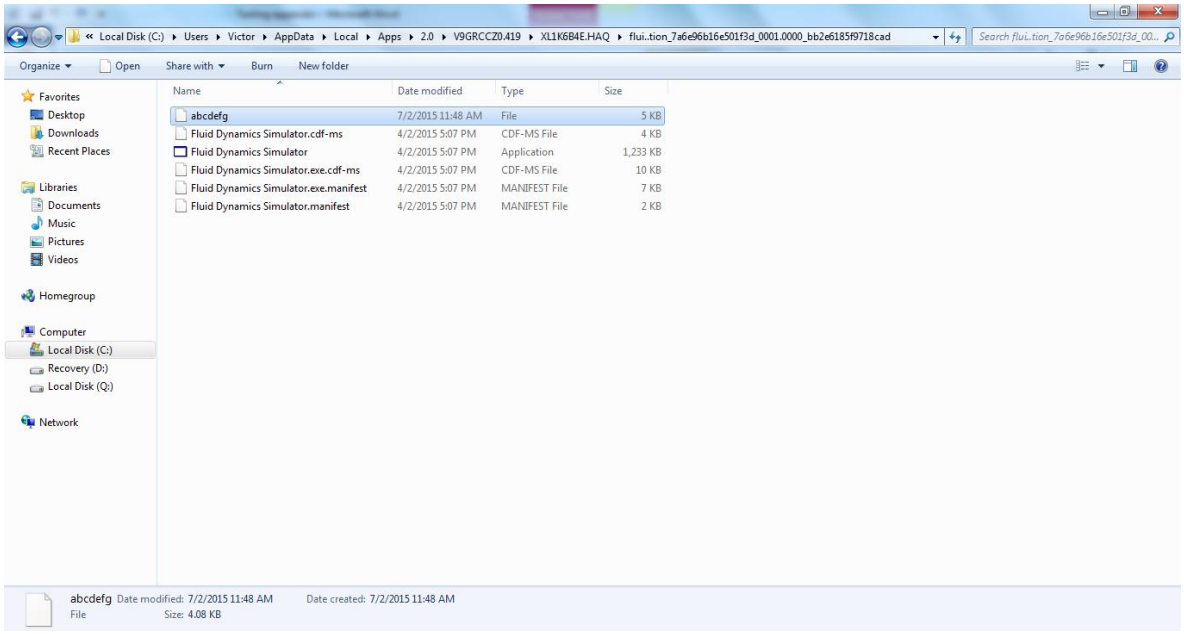


Fig 2.42 (creates a file called abcdefg)

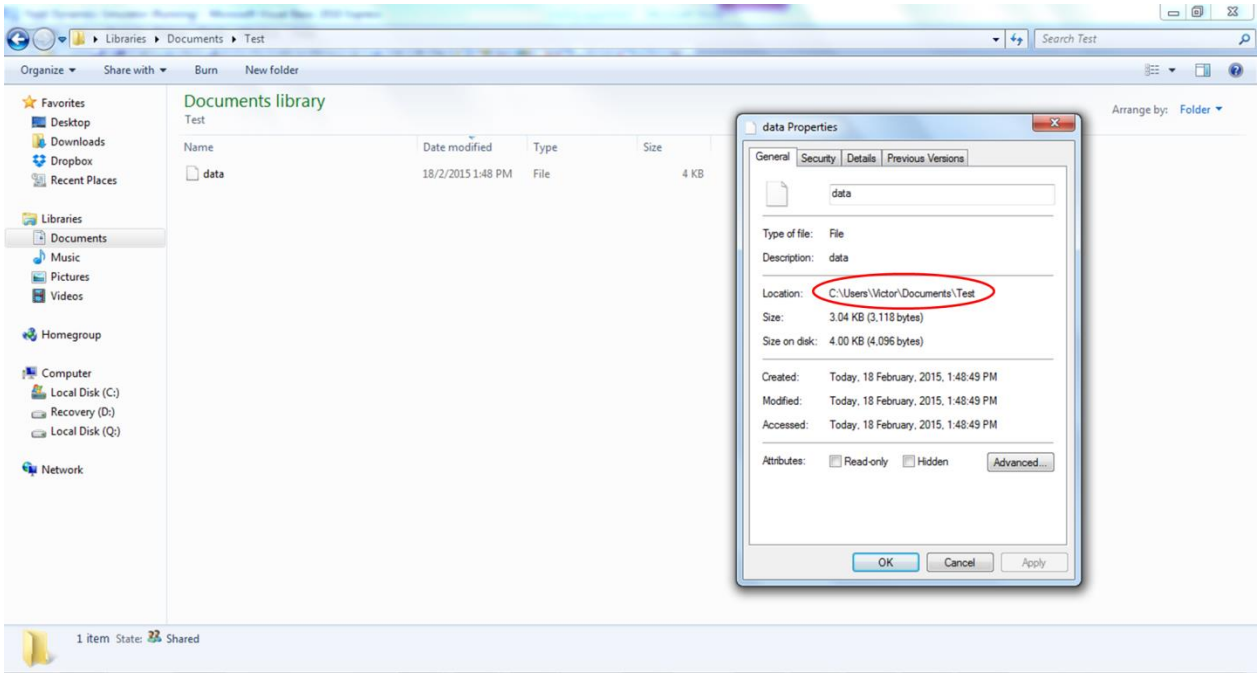


Fig 2.43 (creates a file called data in the right file location but of the wrong format)

White box testing (Trace table)

The reason why the results from simulation and results from calculated spreadsheet look a bit different is because in the simulation, I initialised all the variables to 0 at time 0 when the simulation starts. Using the spreadsheet I made, I took time 0 to be the starting point and had the ball already moving downwards.

Results from simulation					Results from calculated spreadsheet				
Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)	Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)
0	0	0	0	0	0	0	0.033195787	0	7.33788
0.001	0.007338	0.033196	0	7.33788	0.001	0.00733788	0.032023976	0.001171811	7.078852836
0.002	0.014417	0.032024	0.001172	7.078853	0.002	0.014416733	0.030893529	0.002302258	6.828969331
0.003	0.021246	0.030894	0.002302	6.828969	0.003	0.021245702	0.029802988	0.003392799	6.587906714
0.004	0.027834	0.029803	0.003393	6.587907	0.004	0.027833609	0.028750942	0.004444845	6.355353607
0.005	0.034189	0.028751	0.004445	6.355354	0.005	0.034188962	0.027736034	0.005459753	6.131009624
0.006	0.04032	0.027736	0.00546	6.13101	0.006	0.040319972	0.026756952	0.006438835	5.914584984
0.007	0.046235	0.026757	0.006439	5.914585	0.007	0.046234557	0.025812432	0.007383355	5.705800135
0.008	0.05194	0.025812	0.007383	5.7058	0.008	0.051940357	0.024901253	0.008294534	5.50438539
0.009	0.057445	0.024901	0.008295	5.504385	0.009	0.057444743	0.024022239	0.009173548	5.310080586
0.01	0.062755	0.024022	0.009174	5.310081	0.01	0.062754823	0.023174254	0.010021533	5.122634741
0.011	0.067877	0.023174	0.010022	5.122635	0.011	0.067877458	0.022356202	0.010839585	4.941805734
0.012	0.072819	0.022356	0.01084	4.941806	0.012	0.072819264	0.021567029	0.011628759	4.767359992
0.013	0.077587	0.021567	0.011629	4.76736	0.013	0.077586624	0.020805712	0.012390075	4.599072184
0.014	0.082186	0.020806	0.01239	4.599072	0.014	0.082185696	0.020071271	0.013124516	4.436724936
0.015	0.086622	0.020071	0.013125	4.436725	0.015	0.086622421	0.019362755	0.013833032	4.280108546
0.016	0.090903	0.019363	0.013833	4.280109	0.016	0.090902529	0.01867925	0.014516537	4.129020714
0.017	0.095032	0.018679	0.014517	4.129021	0.017	0.09503155	0.018019872	0.015175915	3.983266283
0.018	0.099015	0.01802	0.015176	3.983266	0.018	0.099014816	0.017383771	0.015812016	3.842656983
0.019	0.102857	0.017384	0.015812	3.842657	0.019	0.102857473	0.016770124	0.016425664	3.707011192
0.02	0.106564	0.01677	0.016426	3.707011	0.02	0.106564485	0.016178138	0.017017649	3.576153697
0.021	0.110141	0.016178	0.017018	3.576154	0.021	0.110140638	0.01560705	0.017588737	3.449915471
0.022	0.113591	0.015607	0.017589	3.449916	0.022	0.113590554	0.015056121	0.018139666	3.328133455
0.023	0.116919	0.015056	0.01814	3.328134	0.023	0.116918687	0.01452464	0.018671147	3.210650344
0.024	0.120129	0.014525	0.018671	3.21065	0.024	0.120129337	0.01401192	0.019183867	3.097314387
0.025	0.123227	0.014012	0.019184	3.097315	0.025	0.123226652	0.013517299	0.019678488	2.987979189
0.026	0.126215	0.013517	0.019678	2.987979	0.026	0.126214631	0.013040139	0.020155648	2.882503524

Fig 2.44 (trace table of 6mm radius, 5gcm^{-3} density ball in glycerol)

Results from simulation					Results from calculated spreadsheet				
Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)	Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)
0	0	0	0	0	0	0	0.190951056	0	8.903556
0.001	0.008904	0.190951	0	8.903556	0.001	0.008903556	0.190870499	8.05575E-05	8.899799812
0.002	0.017803	0.19087	8.06E-05	8.8998	0.002	0.017803356	0.190789975	0.000161081	8.896045209
0.003	0.026699	0.19079	0.000161	8.896045	0.003	0.026699401	0.190709486	0.00024157	8.89229219
0.004	0.035592	0.190709	0.000242	8.892292	0.004	0.035591693	0.19062903	0.000322026	8.888540754
0.005	0.04448	0.190629	0.000322	8.888541	0.005	0.044480234	0.190548609	0.000402448	8.884790901
0.006	0.053365	0.190549	0.000402	8.884791	0.006	0.053365025	0.190468221	0.000482835	8.88104263
0.007	0.062246	0.190468	0.000483	8.881043	0.007	0.062246067	0.190387867	0.000563189	8.87729594
0.008	0.071123	0.190388	0.000563	8.877296	0.008	0.071123363	0.190307547	0.000643509	8.873550831
0.009	0.079997	0.190308	0.000644	8.873551	0.009	0.079996914	0.190227261	0.000723795	8.869807302

0.01	0.088867	0.190227	0.000724	8.869807	0.01	0.088866722	0.190147009	0.000804047	8.866065352
0.011	0.097733	0.190147	0.000804	8.866065	0.011	0.097732787	0.190066791	0.000884265	8.862324981
0.012	0.106595	0.190067	0.000884	8.862325	0.012	0.106595112	0.189986606	0.00096445	8.858586187
0.013	0.115454	0.189987	0.000964	8.858586	0.013	0.115453698	0.189906456	0.0010446	8.854848971
0.014	0.124309	0.189906	0.001045	8.854849	0.014	0.124308547	0.189826339	0.001124717	8.851113332
0.015	0.13316	0.189826	0.001125	8.851113	0.015	0.13315966	0.189746256	0.0012048	8.847379268
0.016	0.142007	0.189746	0.001205	8.847379	0.016	0.14200704	0.189666207	0.001284849	8.84364678
0.017	0.150851	0.189666	0.001285	8.843647	0.017	0.150850686	0.189586191	0.001364865	8.839915867
0.018	0.159691	0.189586	0.001365	8.839916	0.018	0.159690602	0.18950621	0.001444847	8.836186527
0.019	0.168527	0.189506	0.001445	8.836187	0.019	0.168526789	0.189426262	0.001524794	8.832458761
0.02	0.177359	0.189426	0.001525	8.832459	0.02	0.177359248	0.189346348	0.001604709	8.828732567

Fig 2.45 (trace table of 8mm radius, 10gcm^{-3} density ball in coconut oil)

Results from simulation					**Results from calculated spreadsheet**				
Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)	Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)
0	0	0	0	0	0	0	0.19229756	0	8.96634
0.001	0.008966	0.192298	0	8.96634	0.001	0.00896634	0.19215221	0.00014535	8.959562708
0.002	0.017926	0.192152	0.000145	8.959563	0.002	0.017925903	0.19200697	0.00029059	8.952790538
0.003	0.026879	0.192007	0.000291	8.952791	0.003	0.026878693	0.19186184	0.00043572	8.946023488
0.004	0.035825	0.191862	0.000436	8.946023	0.004	0.035824717	0.191716819	0.000580741	8.939261552
0.005	0.044764	0.191717	0.000581	8.939262	0.005	0.044763978	0.191571908	0.000725652	8.932504727
0.006	0.053696	0.191572	0.000726	8.932505	0.006	0.053696483	0.191427107	0.000870453	8.92575301
0.007	0.062622	0.191427	0.00087	8.925753	0.007	0.062622236	0.191282415	0.001015145	8.919006396
0.008	0.071541	0.191282	0.001015	8.919006	0.008	0.071541242	0.191137832	0.001159728	8.912264881
0.009	0.080454	0.191138	0.00116	8.912265	0.009	0.080453507	0.190993359	0.001304201	8.905528462
0.01	0.089359	0.190993	0.001304	8.905528	0.01	0.089359036	0.190848995	0.001448565	8.898797135
0.011	0.098258	0.190849	0.001449	8.898797	0.011	0.098257833	0.19070474	0.00159282	8.892070896
0.012	0.10715	0.190705	0.001593	8.892071	0.012	0.107149904	0.190560594	0.001736966	8.885349741
0.013	0.116035	0.190561	0.001737	8.88535	0.013	0.116035254	0.190416557	0.001881003	8.878633666
0.014	0.124914	0.190417	0.001881	8.878634	0.014	0.124913887	0.190272629	0.002024931	8.871922667
0.015	0.133786	0.190273	0.002025	8.871923	0.015	0.13378581	0.190128809	0.002168751	8.865216741
0.016	0.142651	0.190129	0.002169	8.865217	0.016	0.142651027	0.189985099	0.002312461	8.858515884
0.017	0.15151	0.189985	0.002312	8.858516	0.017	0.151509542	0.189841497	0.002456063	8.851820092
0.018	0.160361	0.189841	0.002456	8.85182	0.018	0.160361363	0.189698003	0.002599557	8.845129361
0.019	0.169206	0.189698	0.0026	8.845129	0.019	0.169206492	0.189554618	0.002742942	8.838443687
0.02	0.178045	0.189555	0.002743	8.838444	0.02	0.178044936	0.189411341	0.002886218	8.831763066

Fig 2.46 (trace table of 8mm radius, 10gcm^{-3} density ball in olive oil)

Results from simulation					**Results from calculated spreadsheet**				
Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)	Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)
0	0	0	0	0	0	0	0.000243265	0	3.6297
0.001	0.00363	0.000243	0	3.6297	0.001	0.0036297	5.00517E-05	0.000193213	0.746810775
0.002	0.004377	5.01E-05	0.000193	0.746811	0.002	0.004376511	1.02981E-05	0.000232967	0.153656317

0.003	0.00453	1.03E-05	0.000233	0.153656	0.003	0.004530167	2.11884E-06	0.000241146	0.031614787
0.004	0.004562	2.12E-06	0.000241	0.031615	0.004	0.004561782	4.35952E-07	0.000242829	0.006504742
0.005	0.004568	4.36E-07	0.000243	0.006505	0.005	0.004568287	8.96971E-08	0.000243175	0.001338351
0.006	0.00457	8.97E-08	0.000243	0.001338	0.006	0.004569625	1.84552E-08	0.000243246	0.000275366
0.007	0.00457	1.85E-08	0.000243	0.000275	0.007	0.0045699	3.79715E-09	0.000243261	5.66565E-05
0.008	0.00457	3.80E-09	0.000243	5.67E-05	0.008	0.004569957	7.81264E-10	0.000243264	1.16571E-05
0.009	0.00457	7.81E-10	0.000243	1.17E-05	0.009	0.004569969	1.60745E-10	0.000243265	2.39844E-06
0.01	0.00457	1.61E-10	0.000243	2.40E-06	0.01	0.004569971	3.30733E-11	0.000243265	4.9348E-07
0.011	0.00457	3.31E-11	0.000243	4.93E-07	0.011	0.004569972	6.80484E-12	0.000243265	1.01533E-07
0.012	0.00457	6.80E-12	0.000243	1.02E-07	0.012	0.004569972	1.40009E-12	0.000243265	2.08905E-08
0.013	0.00457	1.40E-12	0.000243	2.09E-08	0.013	0.004569972	2.88069E-13	0.000243265	4.29822E-09
0.014	0.00457	2.88E-13	0.000243	4.30E-09	0.014	0.004569972	5.92704E-14	0.000243265	8.8436E-10
0.015	0.00457	5.93E-14	0.000243	8.84E-10	0.015	0.004569972	1.21948E-14	0.000243265	1.81956E-10
0.016	0.00457	1.22E-14	0.000243	1.82E-10	0.016	0.004569972	2.50917E-15	0.000243265	3.74387E-11
0.017	0.00457	2.51E-15	0.000243	3.74E-11	0.017	0.004569972	5.16189E-16	0.000243265	7.70194E-12
0.018	0.00457	5.16E-16	0.000243	7.70E-12	0.018	0.004569972	1.06143E-16	0.000243265	1.58374E-12
0.019	0.00457	1.06E-16	0.000243	1.58E-12	0.019	0.004569972	2.19009E-17	0.000243265	3.26778E-13
0.02	0.00457	2.19E-17	0.000243	3.27E-13	0.02	0.004569972	4.44523E-18	0.000243265	6.63263E-14
0.021	0.00457	4.55E-18	0.000243	6.79E-14	0.021	0.004569972	9.75782E-19	0.000243265	1.45594E-14
0.022	0.00457	9.76E-19	0.000243	1.46E-14	0.022	0.004569972	0	0.000243265	0
0.023	0.00457	1.08E-19	0.000243	1.62E-15	0.023	0.004569972	0	0.000243265	0
0.024	0.00457	0	0.000243	0	0.024	0.004569972	0	0.000243265	0
0.025	0.00457	0	0.000243	0	0.025	0.004569972	0	0.000243265	0
0.026	0.00457	0	0.000243	0	0.026	0.004569972	0	0.000243265	0
0.027	0.00457	0	0.000243	0	0.027	0.004569972	0	0.000243265	0
0.028	0.00457	0	0.000243	0	0.028	0.004569972	0	0.000243265	0
0.029	0.00457	0	0.000243	0	0.029	0.004569972	0	0.000243265	0
0.03	0.00457	0	0.000243	0	0.03	0.004569972	0	0.000243265	0
0.031	0.00457	0	0.000243	0	0.031	0.004569972	0	0.000243265	0
0.032	0.00457	0	0.000243	0	0.032	0.004569972	0	0.000243265	0
0.033	0.00457	0	0.000243	0	0.033	0.004569972	0	0.000243265	0
0.034	0.00457	0	0.000243	0	0.034	0.004569972	0	0.000243265	0
0.035	0.00457	0	0.000243	0	0.035	0.004569972	0	0.000243265	0
0.036	0.00457	0	0.000243	0	0.036	0.004569972	0	0.000243265	0
0.037	0.00457	0	0.000243	0	0.037	0.004569972	0	0.000243265	0
0.038	0.00457	0	0.000243	0	0.038	0.004569972	0	0.000243265	0
0.039	0.00457	0	0.000243	0	0.039	0.004569972	0	0.000243265	0
0.04	0.00457	0	0.000243	0	0.04	0.004569972	0	0.000243265	0
0.041	0.00457	0	0.000243	0	0.041	0.004569972	0	0.000243265	0
0.042	0.00457	0	0.000243	0	0.042	0.004569972	0	0.000243265	0
0.043	0.00457	0	0.000243	0	0.043	0.004569972	0	0.000243265	0
0.044	0.00457	0	0.000243	0	0.044	0.004569972	0	0.000243265	0
0.045	0.00457	0	0.000243	0	0.045	0.004569972	0	0.000243265	0

0.046	0.00457	0	0.000243	0	0.046	0.004569972	0	0.000243265	0
0.047	0.00457	0	0.000243	0	0.047	0.004569972	0	0.000243265	0
0.048	0.00457	0	0.000243	0	0.048	0.004569972	0	0.000243265	0
0.049	0.00457	0	0.000243	0	0.049	0.004569972	0	0.000243265	0
0.05	0.00457	0	0.000243	0	0.05	0.004569972	0	0.000243265	0
0.051	0.00457	0	0.000243	0	0.051	0.004569972	0	0.000243265	0
0.052	0.00457	0	0.000243	0	0.052	0.004569972	0	0.000243265	0
0.053	0.00457	0	0.000243	0	0.053	0.004569972	0	0.000243265	0
0.054	0.00457	0	0.000243	0	0.054	0.004569972	0	0.000243265	0
0.055	0.00457	0	0.000243	0	0.055	0.004569972	0	0.000243265	0
0.056	0.00457	0	0.000243	0	0.056	0.004569972	0	0.000243265	0
0.057	0.00457	0	0.000243	0	0.057	0.004569972	0	0.000243265	0
0.058	0.00457	0	0.000243	0	0.058	0.004569972	0	0.000243265	0
0.059	0.00457	0	0.000243	0	0.059	0.004569972	0	0.000243265	0
0.06	0.00457	0	0.000243	0	0.06	0.004569972	0	0.000243265	0
0.061	0.00457	0	0.000243	0	0.061	0.004569972	0	0.000243265	0
0.062	0.00457	0	0.000243	0	0.062	0.004569972	0	0.000243265	0
0.063	0.00457	0	0.000243	0	0.063	0.004569972	0	0.000243265	0
0.064	0.00457	0	0.000243	0	0.064	0.004569972	0	0.000243265	0
0.065	0.00457	0	0.000243	0	0.065	0.004569972	0	0.000243265	0
0.066	0.00457	0	0.000243	0	0.066	0.004569972	0	0.000243265	0
0.067	0.00457	0	0.000243	0	0.067	0.004569972	0	0.000243265	0
0.068	0.00457	0	0.000243	0	0.068	0.004569972	0	0.000243265	0
0.069	0.00457	0	0.000243	0	0.069	0.004569972	0	0.000243265	0
0.07	0.00457	0	0.000243	0	0.07	0.004569972	0	0.000243265	0
0.071	0.00457	0	0.000243	0	0.071	0.004569972	0	0.000243265	0
0.072	0.00457	0	0.000243	0	0.072	0.004569972	0	0.000243265	0
0.073	0.00457	0	0.000243	0	0.073	0.004569972	0	0.000243265	0
0.074	0.00457	0	0.000243	0	0.074	0.004569972	0	0.000243265	0
0.075	0.00457	0	0.000243	0	0.075	0.004569972	0	0.000243265	0
0.076	0.00457	0	0.000243	0	0.076	0.004569972	0	0.000243265	0
0.077	0.00457	0	0.000243	0	0.077	0.004569972	0	0.000243265	0
0.078	0.00457	0	0.000243	0	0.078	0.004569972	0	0.000243265	0
0.079	0.00457	0	0.000243	0	0.079	0.004569972	0	0.000243265	0
0.08	0.00457	0	0.000243	0	0.08	0.004569972	0	0.000243265	0
0.081	0.00457	0	0.000243	0	0.081	0.004569972	0	0.000243265	0
0.082	0.00457	0	0.000243	0	0.082	0.004569972	0	0.000243265	0
0.083	0.00457	0	0.000243	0	0.083	0.004569972	0	0.000243265	0
0.084	0.00457	0	0.000243	0	0.084	0.004569972	0	0.000243265	0
0.085	0.00457	0	0.000243	0	0.085	0.004569972	0	0.000243265	0
0.086	0.00457	0	0.000243	0	0.086	0.004569972	0	0.000243265	0
0.087	0.00457	0	0.000243	0	0.087	0.004569972	0	0.000243265	0
0.088	0.00457	0	0.000243	0	0.088	0.004569972	0	0.000243265	0

0.089	0.00457	0	0.000243	0	0.089	0.004569972	0	0.000243265	0
0.09	0.00457	0	0.000243	0	0.09	0.004569972	0	0.000243265	0
0.091	0.00457	0	0.000243	0	0.091	0.004569972	0	0.000243265	0
0.092	0.00457	0	0.000243	0	0.092	0.004569972	0	0.000243265	0
0.093	0.00457	0	0.000243	0	0.093	0.004569972	0	0.000243265	0
0.094	0.00457	0	0.000243	0	0.094	0.004569972	0	0.000243265	0
0.095	0.00457	0	0.000243	0	0.095	0.004569972	0	0.000243265	0
0.096	0.00457	0	0.000243	0	0.096	0.004569972	0	0.000243265	0
0.097	0.00457	0	0.000243	0	0.097	0.004569972	0	0.000243265	0
0.098	0.00457	0	0.000243	0	0.098	0.004569972	0	0.000243265	0
0.099	0.00457	0	0.000243	0	0.099	0.004569972	0	0.000243265	0
0.1	0.00457	0	0.000243	0	0.1	0.004569972	0	0.000243265	0
0.101	0.00457	0	0.000243	0	0.101	0.004569972	0	0.000243265	0
0.102	0.00457	0	0.000243	0	0.102	0.004569972	0	0.000243265	0
0.103	0.00457	0	0.000243	0	0.103	0.004569972	0	0.000243265	0
0.104	0.00457	0	0.000243	0	0.104	0.004569972	0	0.000243265	0
0.105	0.00457	0	0.000243	0	0.105	0.004569972	0	0.000243265	0
0.106	0.00457	0	0.000243	0	0.106	0.004569972	0	0.000243265	0
0.107	0.00457	0	0.000243	0	0.107	0.004569972	0	0.000243265	0
0.108	0.00457	0	0.000243	0	0.108	0.004569972	0	0.000243265	0
0.109	0.00457	0	0.000243	0	0.109	0.004569972	0	0.000243265	0
0.11	0.00457	0	0.000243	0	0.11	0.004569972	0	0.000243265	0
0.111	0.00457	0	0.000243	0	0.111	0.004569972	0	0.000243265	0
0.112	0.00457	0	0.000243	0	0.112	0.004569972	0	0.000243265	0
0.113	0.00457	0	0.000243	0	0.113	0.004569972	0	0.000243265	0
0.114	0.00457	0	0.000243	0	0.114	0.004569972	0	0.000243265	0
0.115	0.00457	0	0.000243	0	0.115	0.004569972	0	0.000243265	0
0.116	0.00457	0	0.000243	0	0.116	0.004569972	0	0.000243265	0
0.117	0.00457	0	0.000243	0	0.117	0.004569972	0	0.000243265	0
0.118	0.00457	0	0.000243	0	0.118	0.004569972	0	0.000243265	0
0.119	0.00457	0	0.000243	0	0.119	0.004569972	0	0.000243265	0
0.12	0.00457	0	0.000243	0	0.12	0.004569972	0	0.000243265	0
0.121	0.00457	0	0.000243	0	0.121	0.004569972	0	0.000243265	0
0.122	0.00457	0	0.000243	0	0.122	0.004569972	0	0.000243265	0
0.123	0.00457	0	0.000243	0	0.123	0.004569972	0	0.000243265	0
0.124	0.00457	0	0.000243	0	0.124	0.004569972	0	0.000243265	0
0.125	0.00457	0	0.000243	0	0.125	0.004569972	0	0.000243265	0
0.126	0.00457	0	0.000243	0	0.126	0.004569972	0	0.000243265	0
0.127	0.00457	0	0.000243	0	0.127	0.004569972	0	0.000243265	0
0.128	0.00457	0	0.000243	0	0.128	0.004569972	0	0.000243265	0
0.129	0.00457	0	0.000243	0	0.129	0.004569972	0	0.000243265	0
0.13	0.00457	0	0.000243	0	0.13	0.004569972	0	0.000243265	0
0.131	0.00457	0	0.000243	0	0.131	0.004569972	0	0.000243265	0

0.132	0.00457	0	0.000243	0	0.132	0.004569972	0	0.000243265	0
0.133	0.00457	0	0.000243	0	0.133	0.004569972	0	0.000243265	0
0.134	0.00457	0	0.000243	0	0.134	0.004569972	0	0.000243265	0
0.135	0.00457	0	0.000243	0	0.135	0.004569972	0	0.000243265	0
0.136	0.00457	0	0.000243	0	0.136	0.004569972	0	0.000243265	0
0.137	0.00457	0	0.000243	0	0.137	0.004569972	0	0.000243265	0
0.138	0.00457	0	0.000243	0	0.138	0.004569972	0	0.000243265	0
0.139	0.00457	0	0.000243	0	0.139	0.004569972	0	0.000243265	0
0.14	0.00457	0	0.000243	0	0.14	0.004569972	0	0.000243265	0
0.141	0.00457	0	0.000243	0	0.141	0.004569972	0	0.000243265	0
0.142	0.00457	0	0.000243	0	0.142	0.004569972	0	0.000243265	0
0.143	0.00457	0	0.000243	0	0.143	0.004569972	0	0.000243265	0
0.144	0.00457	0	0.000243	0	0.144	0.004569972	0	0.000243265	0
0.145	0.00457	0	0.000243	0	0.145	0.004569972	0	0.000243265	0
0.146	0.00457	0	0.000243	0	0.146	0.004569972	0	0.000243265	0
0.147	0.00457	0	0.000243	0	0.147	0.004569972	0	0.000243265	0
0.148	0.00457	0	0.000243	0	0.148	0.004569972	0	0.000243265	0
0.149	0.00457	0	0.000243	0	0.149	0.004569972	0	0.000243265	0
0.15	0.00457	0	0.000243	0	0.15	0.004569972	0	0.000243265	0
0.151	0.00457	0	0.000243	0	0.151	0.004569972	0	0.000243265	0
0.152	0.00457	0	0.000243	0	0.152	0.004569972	0	0.000243265	0
0.153	0.00457	0	0.000243	0	0.153	0.004569972	0	0.000243265	0
0.154	0.00457	0	0.000243	0	0.154	0.004569972	0	0.000243265	0
0.155	0.00457	0	0.000243	0	0.155	0.004569972	0	0.000243265	0
0.156	0.00457	0	0.000243	0	0.156	0.004569972	0	0.000243265	0
0.157	0.00457	0	0.000243	0	0.157	0.004569972	0	0.000243265	0
0.158	0.00457	0	0.000243	0	0.158	0.004569972	0	0.000243265	0
0.159	0.00457	0	0.000243	0	0.159	0.004569972	0	0.000243265	0
0.16	0.00457	0	0.000243	0	0.16	0.004569972	0	0.000243265	0
0.161	0.00457	0	0.000243	0	0.161	0.004569972	0	0.000243265	0
0.162	0.00457	0	0.000243	0	0.162	0.004569972	0	0.000243265	0
0.163	0.00457	0	0.000243	0	0.163	0.004569972	0	0.000243265	0
0.164	0.00457	0	0.000243	0	0.164	0.004569972	0	0.000243265	0
0.165	0.00457	0	0.000243	0	0.165	0.004569972	0	0.000243265	0
0.166	0.00457	0	0.000243	0	0.166	0.004569972	0	0.000243265	0
0.167	0.00457	0	0.000243	0	0.167	0.004569972	0	0.000243265	0
0.168	0.00457	0	0.000243	0	0.168	0.004569972	0	0.000243265	0
0.169	0.00457	0	0.000243	0	0.169	0.004569972	0	0.000243265	0
0.17	0.00457	0	0.000243	0	0.17	0.004569972	0	0.000243265	0
0.171	0.00457	0	0.000243	0	0.171	0.004569972	0	0.000243265	0
0.172	0.00457	0	0.000243	0	0.172	0.004569972	0	0.000243265	0
0.173	0.00457	0	0.000243	0	0.173	0.004569972	0	0.000243265	0
0.174	0.00457	0	0.000243	0	0.174	0.004569972	0	0.000243265	0

0.175	0.00457	0	0.000243	0	0.175	0.004569972	0	0.000243265	0
0.176	0.00457	0	0.000243	0	0.176	0.004569972	0	0.000243265	0
0.177	0.00457	0	0.000243	0	0.177	0.004569972	0	0.000243265	0
0.178	0.00457	0	0.000243	0	0.178	0.004569972	0	0.000243265	0
0.179	0.00457	0	0.000243	0	0.179	0.004569972	0	0.000243265	0
0.18	0.00457	0	0.000243	0	0.18	0.004569972	0	0.000243265	0
0.181	0.00457	0	0.000243	0	0.181	0.004569972	0	0.000243265	0
0.182	0.00457	0	0.000243	0	0.182	0.004569972	0	0.000243265	0
0.183	0.00457	0	0.000243	0	0.183	0.004569972	0	0.000243265	0
0.184	0.00457	0	0.000243	0	0.184	0.004569972	0	0.000243265	0
0.185	0.00457	0	0.000243	0	0.185	0.004569972	0	0.000243265	0
0.186	0.00457	0	0.000243	0	0.186	0.004569972	0	0.000243265	0
0.187	0.00457	0	0.000243	0	0.187	0.004569972	0	0.000243265	0
0.188	0.00457	0	0.000243	0	0.188	0.004569972	0	0.000243265	0
0.189	0.00457	0	0.000243	0	0.189	0.004569972	0	0.000243265	0
0.19	0.00457	0	0.000243	0	0.19	0.004569972	0	0.000243265	0
0.191	0.00457	0	0.000243	0	0.191	0.004569972	0	0.000243265	0
0.192	0.00457	0	0.000243	0	0.192	0.004569972	0	0.000243265	0
0.193	0.00457	0	0.000243	0	0.193	0.004569972	0	0.000243265	0
0.194	0.00457	0	0.000243	0	0.194	0.004569972	0	0.000243265	0
0.195	0.00457	0	0.000243	0	0.195	0.004569972	0	0.000243265	0
0.196	0.00457	0	0.000243	0	0.196	0.004569972	0	0.000243265	0
0.197	0.00457	0	0.000243	0	0.197	0.004569972	0	0.000243265	0
0.198	0.00457	0	0.000243	0	0.198	0.004569972	0	0.000243265	0
0.199	0.00457	0	0.000243	0	0.199	0.004569972	0	0.000243265	0
0.2	0.00457	0	0.000243	0	0.2	0.004569972	0	0.000243265	0
0.201	0.00457	0	0.000243	0	0.201	0.004569972	0	0.000243265	0
0.202	0.00457	0	0.000243	0	0.202	0.004569972	0	0.000243265	0
0.203	0.00457	0	0.000243	0	0.203	0.004569972	0	0.000243265	0
0.204	0.00457	0	0.000243	0	0.204	0.004569972	0	0.000243265	0
0.205	0.00457	0	0.000243	0	0.205	0.004569972	0	0.000243265	0
0.206	0.00457	0	0.000243	0	0.206	0.004569972	0	0.000243265	0
0.207	0.00457	0	0.000243	0	0.207	0.004569972	0	0.000243265	0
0.208	0.00457	0	0.000243	0	0.208	0.004569972	0	0.000243265	0
0.209	0.00457	0	0.000243	0	0.209	0.004569972	0	0.000243265	0
0.21	0.00457	0	0.000243	0	0.21	0.004569972	0	0.000243265	0
0.211	0.00457	0	0.000243	0	0.211	0.004569972	0	0.000243265	0
0.212	0.00457	0	0.000243	0	0.212	0.004569972	0	0.000243265	0
0.213	0.00457	0	0.000243	0	0.213	0.004569972	0	0.000243265	0
0.214	0.00457	0	0.000243	0	0.214	0.004569972	0	0.000243265	0
0.215	0.00457	0	0.000243	0	0.215	0.004569972	0	0.000243265	0
0.216	0.00457	0	0.000243	0	0.216	0.004569972	0	0.000243265	0
0.217	0.00457	0	0.000243	0	0.217	0.004569972	0	0.000243265	0

0.218	0.00457	0	0.000243	0	0.218	0.004569972	0	0.000243265	0
0.219	0.00457	0	0.000243	0	0.219	0.004569972	0	0.000243265	0
0.22	0.00457	0	0.000243	0	0.22	0.004569972	0	0.000243265	0
0.221	0.00457	0	0.000243	0	0.221	0.004569972	0	0.000243265	0
0.222	0.00457	0	0.000243	0	0.222	0.004569972	0	0.000243265	0
0.223	0.00457	0	0.000243	0	0.223	0.004569972	0	0.000243265	0
0.224	0.00457	0	0.000243	0	0.224	0.004569972	0	0.000243265	0
0.225	0.00457	0	0.000243	0	0.225	0.004569972	0	0.000243265	0
0.226	0.00457	0	0.000243	0	0.226	0.004569972	0	0.000243265	0
0.227	0.00457	0	0.000243	0	0.227	0.004569972	0	0.000243265	0
0.228	0.00457	0	0.000243	0	0.228	0.004569972	0	0.000243265	0
0.229	0.00457	0	0.000243	0	0.229	0.004569972	0	0.000243265	0
0.23	0.00457	0	0.000243	0	0.23	0.004569972	0	0.000243265	0
0.231	0.00457	0	0.000243	0	0.231	0.004569972	0	0.000243265	0
0.232	0.00457	0	0.000243	0	0.232	0.004569972	0	0.000243265	0
0.233	0.00457	0	0.000243	0	0.233	0.004569972	0	0.000243265	0
0.234	0.00457	0	0.000243	0	0.234	0.004569972	0	0.000243265	0
0.235	0.00457	0	0.000243	0	0.235	0.004569972	0	0.000243265	0
0.236	0.00457	0	0.000243	0	0.236	0.004569972	0	0.000243265	0
0.237	0.00457	0	0.000243	0	0.237	0.004569972	0	0.000243265	0
0.238	0.00457	0	0.000243	0	0.238	0.004569972	0	0.000243265	0
0.239	0.00457	0	0.000243	0	0.239	0.004569972	0	0.000243265	0
0.24	0.00457	0	0.000243	0	0.24	0.004569972	0	0.000243265	0
0.241	0.00457	0	0.000243	0	0.241	0.004569972	0	0.000243265	0
0.242	0.00457	0	0.000243	0	0.242	0.004569972	0	0.000243265	0
0.243	0.00457	0	0.000243	0	0.243	0.004569972	0	0.000243265	0
0.244	0.00457	0	0.000243	0	0.244	0.004569972	0	0.000243265	0
0.245	0.00457	0	0.000243	0	0.245	0.004569972	0	0.000243265	0
0.246	0.00457	0	0.000243	0	0.246	0.004569972	0	0.000243265	0
0.247	0.00457	0	0.000243	0	0.247	0.004569972	0	0.000243265	0
0.248	0.00457	0	0.000243	0	0.248	0.004569972	0	0.000243265	0
0.249	0.00457	0	0.000243	0	0.249	0.004569972	0	0.000243265	0
0.25	0.00457	0	0.000243	0	0.25	0.004569972	0	0.000243265	0
0.251	0.00457	0	0.000243	0	0.251	0.004569972	0	0.000243265	0
0.252	0.00457	0	0.000243	0	0.252	0.004569972	0	0.000243265	0
0.253	0.00457	0	0.000243	0	0.253	0.004569972	0	0.000243265	0
0.254	0.00457	0	0.000243	0	0.254	0.004569972	0	0.000243265	0
0.255	0.00457	0	0.000243	0	0.255	0.004569972	0	0.000243265	0
0.256	0.00457	0	0.000243	0	0.256	0.004569972	0	0.000243265	0
0.257	0.00457	0	0.000243	0	0.257	0.004569972	0	0.000243265	0
0.258	0.00457	0	0.000243	0	0.258	0.004569972	0	0.000243265	0
0.259	0.00457	0	0.000243	0	0.259	0.004569972	0	0.000243265	0
0.26	0.00457	0	0.000243	0	0.26	0.004569972	0	0.000243265	0

0.261	0.00457	0	0.000243	0	0.261	0.004569972	0	0.000243265	0
0.262	0.00457	0	0.000243	0	0.262	0.004569972	0	0.000243265	0
0.263	0.00457	0	0.000243	0	0.263	0.004569972	0	0.000243265	0
0.264	0.00457	0	0.000243	0	0.264	0.004569972	0	0.000243265	0
0.265	0.00457	0	0.000243	0	0.265	0.004569972	0	0.000243265	0
0.266	0.00457	0	0.000243	0	0.266	0.004569972	0	0.000243265	0
0.267	0.00457	0	0.000243	0	0.267	0.004569972	0	0.000243265	0
0.268	0.00457	0	0.000243	0	0.268	0.004569972	0	0.000243265	0
0.269	0.00457	0	0.000243	0	0.269	0.004569972	0	0.000243265	0
0.27	0.00457	0	0.000243	0	0.27	0.004569972	0	0.000243265	0
0.271	0.00457	0	0.000243	0	0.271	0.004569972	0	0.000243265	0
0.272	0.00457	0	0.000243	0	0.272	0.004569972	0	0.000243265	0
0.273	0.00457	0	0.000243	0	0.273	0.004569972	0	0.000243265	0
0.274	0.00457	0	0.000243	0	0.274	0.004569972	0	0.000243265	0
0.275	0.00457	0	0.000243	0	0.275	0.004569972	0	0.000243265	0
0.276	0.00457	0	0.000243	0	0.276	0.004569972	0	0.000243265	0
0.277	0.00457	0	0.000243	0	0.277	0.004569972	0	0.000243265	0
0.278	0.00457	0	0.000243	0	0.278	0.004569972	0	0.000243265	0
0.279	0.00457	0	0.000243	0	0.279	0.004569972	0	0.000243265	0
0.28	0.00457	0	0.000243	0	0.28	0.004569972	0	0.000243265	0
0.281	0.00457	0	0.000243	0	0.281	0.004569972	0	0.000243265	0
0.282	0.00457	0	0.000243	0	0.282	0.004569972	0	0.000243265	0
0.283	0.00457	0	0.000243	0	0.283	0.004569972	0	0.000243265	0
0.284	0.00457	0	0.000243	0	0.284	0.004569972	0	0.000243265	0
0.285	0.00457	0	0.000243	0	0.285	0.004569972	0	0.000243265	0
0.286	0.00457	0	0.000243	0	0.286	0.004569972	0	0.000243265	0
0.287	0.00457	0	0.000243	0	0.287	0.004569972	0	0.000243265	0
0.288	0.00457	0	0.000243	0	0.288	0.004569972	0	0.000243265	0
0.289	0.00457	0	0.000243	0	0.289	0.004569972	0	0.000243265	0
0.29	0.00457	0	0.000243	0	0.29	0.004569972	0	0.000243265	0
0.291	0.00457	0	0.000243	0	0.291	0.004569972	0	0.000243265	0
0.292	0.00457	0	0.000243	0	0.292	0.004569972	0	0.000243265	0
0.293	0.00457	0	0.000243	0	0.293	0.004569972	0	0.000243265	0
0.294	0.00457	0	0.000243	0	0.294	0.004569972	0	0.000243265	0
0.295	0.00457	0	0.000243	0	0.295	0.004569972	0	0.000243265	0
0.296	0.00457	0	0.000243	0	0.296	0.004569972	0	0.000243265	0
0.297	0.00457	0	0.000243	0	0.297	0.004569972	0	0.000243265	0
0.298	0.00457	0	0.000243	0	0.298	0.004569972	0	0.000243265	0
0.299	0.00457	0	0.000243	0	0.299	0.004569972	0	0.000243265	0
0.3	0.00457	0	0.000243	0	0.3	0.004569972	0	0.000243265	0
0.301	0.00457	0	0.000243	0	0.301	0.004569972	0	0.000243265	0
0.302	0.00457	0	0.000243	0	0.302	0.004569972	0	0.000243265	0
0.303	0.00457	0	0.000243	0	0.303	0.004569972	0	0.000243265	0

0.304	0.00457	0	0.000243	0	0.304	0.004569972	0	0.000243265	0
0.305	0.00457	0	0.000243	0	0.305	0.004569972	0	0.000243265	0
0.306	0.00457	0	0.000243	0	0.306	0.004569972	0	0.000243265	0
0.307	0.00457	0	0.000243	0	0.307	0.004569972	0	0.000243265	0
0.308	0.00457	0	0.000243	0	0.308	0.004569972	0	0.000243265	0
0.309	0.00457	0	0.000243	0	0.309	0.004569972	0	0.000243265	0
0.31	0.00457	0	0.000243	0	0.31	0.004569972	0	0.000243265	0
0.311	0.00457	0	0.000243	0	0.311	0.004569972	0	0.000243265	0
0.312	0.00457	0	0.000243	0	0.312	0.004569972	0	0.000243265	0
0.313	0.00457	0	0.000243	0	0.313	0.004569972	0	0.000243265	0
0.314	0.00457	0	0.000243	0	0.314	0.004569972	0	0.000243265	0
0.315	0.00457	0	0.000243	0	0.315	0.004569972	0	0.000243265	0
0.316	0.00457	0	0.000243	0	0.316	0.004569972	0	0.000243265	0
0.317	0.00457	0	0.000243	0	0.317	0.004569972	0	0.000243265	0
0.318	0.00457	0	0.000243	0	0.318	0.004569972	0	0.000243265	0
0.318999	0.00457	0	0.000243	0	0.319	0.004569972	0	0.000243265	0
0.319999	0.00457	0	0.000243	0	0.32	0.004569972	0	0.000243265	0
0.320999	0.00457	0	0.000243	0	0.321	0.004569972	0	0.000243265	0
0.321999	0.00457	0	0.000243	0	0.322	0.004569972	0	0.000243265	0
0.322999	0.00457	0	0.000243	0	0.323	0.004569972	0	0.000243265	0
0.323999	0.00457	0	0.000243	0	0.324	0.004569972	0	0.000243265	0
0.324999	0.00457	0	0.000243	0	0.325	0.004569972	0	0.000243265	0
0.325999	0.00457	0	0.000243	0	0.326	0.004569972	0	0.000243265	0
0.326999	0.00457	0	0.000243	0	0.327	0.004569972	0	0.000243265	0
0.327999	0.00457	0	0.000243	0	0.328	0.004569972	0	0.000243265	0
0.328999	0.00457	0	0.000243	0	0.329	0.004569972	0	0.000243265	0
0.329999	0.00457	0	0.000243	0	0.33	0.004569972	0	0.000243265	0
0.330999	0.00457	0	0.000243	0	0.331	0.004569972	0	0.000243265	0
0.331999	0.00457	0	0.000243	0	0.332	0.004569972	0	0.000243265	0
0.332999	0.00457	0	0.000243	0	0.333	0.004569972	0	0.000243265	0
0.333999	0.00457	0	0.000243	0	0.334	0.004569972	0	0.000243265	0
0.334999	0.00457	0	0.000243	0	0.335	0.004569972	0	0.000243265	0
0.335999	0.00457	0	0.000243	0	0.336	0.004569972	0	0.000243265	0
0.336999	0.00457	0	0.000243	0	0.337	0.004569972	0	0.000243265	0
0.337999	0.00457	0	0.000243	0	0.338	0.004569972	0	0.000243265	0
0.338999	0.00457	0	0.000243	0	0.339	0.004569972	0	0.000243265	0
0.339999	0.00457	0	0.000243	0	0.34	0.004569972	0	0.000243265	0
0.340999	0.00457	0	0.000243	0	0.341	0.004569972	0	0.000243265	0
0.341999	0.00457	0	0.000243	0	0.342	0.004569972	0	0.000243265	0
0.342999	0.00457	0	0.000243	0	0.343	0.004569972	0	0.000243265	0
0.343999	0.00457	0	0.000243	0	0.344	0.004569972	0	0.000243265	0
0.344999	0.00457	0	0.000243	0	0.345	0.004569972	0	0.000243265	0
0.345999	0.00457	0	0.000243	0	0.346	0.004569972	0	0.000243265	0

0.346999	0.00457	0	0.000243	0	0.347	0.004569972	0	0.000243265	0
0.347999	0.00457	0	0.000243	0	0.348	0.004569972	0	0.000243265	0
0.348999	0.00457	0	0.000243	0	0.349	0.004569972	0	0.000243265	0
0.349999	0.00457	0	0.000243	0	0.35	0.004569972	0	0.000243265	0
0.350999	0.00457	0	0.000243	0	0.351	0.004569972	0	0.000243265	0
0.351999	0.00457	0	0.000243	0	0.352	0.004569972	0	0.000243265	0
0.352999	0.00457	0	0.000243	0	0.353	0.004569972	0	0.000243265	0
0.353999	0.00457	0	0.000243	0	0.354	0.004569972	0	0.000243265	0
0.354999	0.00457	0	0.000243	0	0.355	0.004569972	0	0.000243265	0
0.355999	0.00457	0	0.000243	0	0.356	0.004569972	0	0.000243265	0
0.356999	0.00457	0	0.000243	0	0.357	0.004569972	0	0.000243265	0
0.357999	0.00457	0	0.000243	0	0.358	0.004569972	0	0.000243265	0
0.358999	0.00457	0	0.000243	0	0.359	0.004569972	0	0.000243265	0
0.359999	0.00457	0	0.000243	0	0.36	0.004569972	0	0.000243265	0
0.360999	0.00457	0	0.000243	0	0.361	0.004569972	0	0.000243265	0
0.361999	0.00457	0	0.000243	0	0.362	0.004569972	0	0.000243265	0
0.362999	0.00457	0	0.000243	0	0.363	0.004569972	0	0.000243265	0
0.363999	0.00457	0	0.000243	0	0.364	0.004569972	0	0.000243265	0
0.364999	0.00457	0	0.000243	0	0.365	0.004569972	0	0.000243265	0
0.365999	0.00457	0	0.000243	0	0.366	0.004569972	0	0.000243265	0
0.366999	0.00457	0	0.000243	0	0.367	0.004569972	0	0.000243265	0
0.367999	0.00457	0	0.000243	0	0.368	0.004569972	0	0.000243265	0
0.368999	0.00457	0	0.000243	0	0.369	0.004569972	0	0.000243265	0
0.369999	0.00457	0	0.000243	0	0.37	0.004569972	0	0.000243265	0
0.370999	0.00457	0	0.000243	0	0.371	0.004569972	0	0.000243265	0
0.371999	0.00457	0	0.000243	0	0.372	0.004569972	0	0.000243265	0
0.372999	0.00457	0	0.000243	0	0.373	0.004569972	0	0.000243265	0
0.373999	0.00457	0	0.000243	0	0.374	0.004569972	0	0.000243265	0
0.374999	0.00457	0	0.000243	0	0.375	0.004569972	0	0.000243265	0
0.375999	0.00457	0	0.000243	0	0.376	0.004569972	0	0.000243265	0
0.376999	0.00457	0	0.000243	0	0.377	0.004569972	0	0.000243265	0
0.377999	0.00457	0	0.000243	0	0.378	0.004569972	0	0.000243265	0
0.378999	0.00457	0	0.000243	0	0.379	0.004569972	0	0.000243265	0
0.379999	0.00457	0	0.000243	0	0.38	0.004569972	0	0.000243265	0
0.380999	0.00457	0	0.000243	0	0.381	0.004569972	0	0.000243265	0
0.381999	0.00457	0	0.000243	0	0.382	0.004569972	0	0.000243265	0
0.382999	0.00457	0	0.000243	0	0.383	0.004569972	0	0.000243265	0
0.383999	0.00457	0	0.000243	0	0.384	0.004569972	0	0.000243265	0
0.384999	0.00457	0	0.000243	0	0.385	0.004569972	0	0.000243265	0
0.385999	0.00457	0	0.000243	0	0.386	0.004569972	0	0.000243265	0
0.386999	0.00457	0	0.000243	0	0.387	0.004569972	0	0.000243265	0
0.387999	0.00457	0	0.000243	0	0.388	0.004569972	0	0.000243265	0
0.388999	0.00457	0	0.000243	0	0.389	0.004569972	0	0.000243265	0

0.389999	0.00457	0	0.000243	0	0.39	0.004569972	0	0.000243265	0
0.390999	0.00457	0	0.000243	0	0.391	0.004569972	0	0.000243265	0
0.391999	0.00457	0	0.000243	0	0.392	0.004569972	0	0.000243265	0
0.392999	0.00457	0	0.000243	0	0.393	0.004569972	0	0.000243265	0
0.393999	0.00457	0	0.000243	0	0.394	0.004569972	0	0.000243265	0
0.394999	0.00457	0	0.000243	0	0.395	0.004569972	0	0.000243265	0
0.395998	0.00457	0	0.000243	0	0.396	0.004569972	0	0.000243265	0
0.396998	0.00457	0	0.000243	0	0.397	0.004569972	0	0.000243265	0
0.397998	0.00457	0	0.000243	0	0.398	0.004569972	0	0.000243265	0
0.398998	0.00457	0	0.000243	0	0.399	0.004569972	0	0.000243265	0
0.399998	0.00457	0	0.000243	0	0.4	0.004569972	0	0.000243265	0
0.400998	0.00457	0	0.000243	0	0.401	0.004569972	0	0.000243265	0
0.401998	0.00457	0	0.000243	0	0.402	0.004569972	0	0.000243265	0
0.402998	0.00457	0	0.000243	0	0.403	0.004569972	0	0.000243265	0
0.403998	0.00457	0	0.000243	0	0.404	0.004569972	0	0.000243265	0
0.404998	0.00457	0	0.000243	0	0.405	0.004569972	0	0.000243265	0
0.405998	0.00457	0	0.000243	0	0.406	0.004569972	0	0.000243265	0
0.406998	0.00457	0	0.000243	0	0.407	0.004569972	0	0.000243265	0
0.407998	0.00457	0	0.000243	0	0.408	0.004569972	0	0.000243265	0
0.408998	0.00457	0	0.000243	0	0.409	0.004569972	0	0.000243265	0
0.409998	0.00457	0	0.000243	0	0.41	0.004569972	0	0.000243265	0
0.410998	0.00457	0	0.000243	0	0.411	0.004569972	0	0.000243265	0
0.411998	0.00457	0	0.000243	0	0.412	0.004569972	0	0.000243265	0
0.412998	0.00457	0	0.000243	0	0.413	0.004569972	0	0.000243265	0
0.413998	0.00457	0	0.000243	0	0.414	0.004569972	0	0.000243265	0
0.414998	0.00457	0	0.000243	0	0.415	0.004569972	0	0.000243265	0
0.415998	0.00457	0	0.000243	0	0.416	0.004569972	0	0.000243265	0
0.416998	0.00457	0	0.000243	0	0.417	0.004569972	0	0.000243265	0
0.417998	0.00457	0	0.000243	0	0.418	0.004569972	0	0.000243265	0
0.418998	0.00457	0	0.000243	0	0.419	0.004569972	0	0.000243265	0
0.419998	0.00457	0	0.000243	0	0.42	0.004569972	0	0.000243265	0
0.420998	0.00457	0	0.000243	0	0.421	0.004569972	0	0.000243265	0
0.421998	0.00457	0	0.000243	0	0.422	0.004569972	0	0.000243265	0
0.422998	0.00457	0	0.000243	0	0.423	0.004569972	0	0.000243265	0
0.423998	0.00457	0	0.000243	0	0.424	0.004569972	0	0.000243265	0
0.424998	0.00457	0	0.000243	0	0.425	0.004569972	0	0.000243265	0
0.425998	0.00457	0	0.000243	0	0.426	0.004569972	0	0.000243265	0
0.426998	0.00457	0	0.000243	0	0.427	0.004569972	0	0.000243265	0
0.427998	0.00457	0	0.000243	0	0.428	0.004569972	0	0.000243265	0
0.428998	0.00457	0	0.000243	0	0.429	0.004569972	0	0.000243265	0
0.429998	0.00457	0	0.000243	0	0.43	0.004569972	0	0.000243265	0
0.430998	0.00457	0	0.000243	0	0.431	0.004569972	0	0.000243265	0
0.431998	0.00457	0	0.000243	0	0.432	0.004569972	0	0.000243265	0

0.432998	0.00457	0	0.000243	0	0.433	0.004569972	0	0.000243265	0
0.433998	0.00457	0	0.000243	0	0.434	0.004569972	0	0.000243265	0
0.434998	0.00457	0	0.000243	0	0.435	0.004569972	0	0.000243265	0
0.435998	0.00457	0	0.000243	0	0.436	0.004569972	0	0.000243265	0
0.436998	0.00457	0	0.000243	0	0.437	0.004569972	0	0.000243265	0
0.437998	0.00457	0	0.000243	0	0.438	0.004569972	0	0.000243265	0
0.438998	0.00457	0	0.000243	0	0.439	0.004569972	0	0.000243265	0
0.439998	0.00457	0	0.000243	0	0.44	0.004569972	0	0.000243265	0
0.440998	0.00457	0	0.000243	0	0.441	0.004569972	0	0.000243265	0
0.441998	0.00457	0	0.000243	0	0.442	0.004569972	0	0.000243265	0
0.442998	0.00457	0	0.000243	0	0.443	0.004569972	0	0.000243265	0
0.443998	0.00457	0	0.000243	0	0.444	0.004569972	0	0.000243265	0
0.444998	0.00457	0	0.000243	0	0.445	0.004569972	0	0.000243265	0
0.445998	0.00457	0	0.000243	0	0.446	0.004569972	0	0.000243265	0
0.446998	0.00457	0	0.000243	0	0.447	0.004569972	0	0.000243265	0
0.447998	0.00457	0	0.000243	0	0.448	0.004569972	0	0.000243265	0
0.448998	0.00457	0	0.000243	0	0.449	0.004569972	0	0.000243265	0
0.449998	0.00457	0	0.000243	0	0.45	0.004569972	0	0.000243265	0
0.450998	0.00457	0	0.000243	0	0.451	0.004569972	0	0.000243265	0
0.451998	0.00457	0	0.000243	0	0.452	0.004569972	0	0.000243265	0
0.452998	0.00457	0	0.000243	0	0.453	0.004569972	0	0.000243265	0
0.453998	0.00457	0	0.000243	0	0.454	0.004569972	0	0.000243265	0
0.454998	0.00457	0	0.000243	0	0.455	0.004569972	0	0.000243265	0
0.455998	0.00457	0	0.000243	0	0.456	0.004569972	0	0.000243265	0
0.456998	0.00457	0	0.000243	0	0.457	0.004569972	0	0.000243265	0
0.457998	0.00457	0	0.000243	0	0.458	0.004569972	0	0.000243265	0
0.458998	0.00457	0	0.000243	0	0.459	0.004569972	0	0.000243265	0
0.459998	0.00457	0	0.000243	0	0.46	0.004569972	0	0.000243265	0
0.460998	0.00457	0	0.000243	0	0.461	0.004569972	0	0.000243265	0
0.461998	0.00457	0	0.000243	0	0.462	0.004569972	0	0.000243265	0
0.462998	0.00457	0	0.000243	0	0.463	0.004569972	0	0.000243265	0
0.463998	0.00457	0	0.000243	0	0.464	0.004569972	0	0.000243265	0
0.464998	0.00457	0	0.000243	0	0.465	0.004569972	0	0.000243265	0
0.465998	0.00457	0	0.000243	0	0.466	0.004569972	0	0.000243265	0
0.466998	0.00457	0	0.000243	0	0.467	0.004569972	0	0.000243265	0
0.467998	0.00457	0	0.000243	0	0.468	0.004569972	0	0.000243265	0
0.468998	0.00457	0	0.000243	0	0.469	0.004569972	0	0.000243265	0
0.469998	0.00457	0	0.000243	0	0.47	0.004569972	0	0.000243265	0
0.470998	0.00457	0	0.000243	0	0.471	0.004569972	0	0.000243265	0
0.471998	0.00457	0	0.000243	0	0.472	0.004569972	0	0.000243265	0
0.472998	0.00457	0	0.000243	0	0.473	0.004569972	0	0.000243265	0
0.473997	0.00457	0	0.000243	0	0.474	0.004569972	0	0.000243265	0
0.474997	0.00457	0	0.000243	0	0.475	0.004569972	0	0.000243265	0

0.475997	0.00457	0	0.000243	0	0.476	0.004569972	0	0.000243265	0
0.476997	0.00457	0	0.000243	0	0.477	0.004569972	0	0.000243265	0

Fig 2.47 (trace table of 2mm radius, 2gcm^{-3} ball in glycerol)

Results from simulation					Results from calculated spreadsheet				
Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)	Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)
0	0	0	0	0	0	0	0.149798537	0	8.676670807
0.001	0.008677	0.149799	0	8.676671	0.001	0.008676671	0.149155584	0.000642953	8.639429499
0.002	0.017316	0.149156	0.000643	8.639429	0.002	0.0173161	0.14851539	0.001283147	8.602348035
0.003	0.025918	0.148515	0.001283	8.602348	0.003	0.025918448	0.147877944	0.001920593	8.565425728
0.004	0.034484	0.147878	0.001921	8.565426	0.004	0.034483874	0.147243234	0.002555302	8.528661897
0.005	0.043013	0.147243	0.002555	8.528662	0.005	0.043012536	0.146611249	0.003187288	8.49205586
0.006	0.051505	0.146611	0.003187	8.492056	0.006	0.051504592	0.145981976	0.003816561	8.455606941
0.007	0.05996	0.145982	0.003817	8.455607	0.007	0.059960199	0.145355404	0.004443133	8.419314465
0.008	0.06838	0.145355	0.004443	8.419314	0.008	0.068379513	0.144731521	0.005067016	8.383177761
0.009	0.076763	0.144732	0.005067	8.383178	0.009	0.076762691	0.144110316	0.005688221	8.34719616
0.01	0.08511	0.14411	0.005688	8.347196	0.01	0.085109887	0.143491777	0.00630676	8.311368996
0.011	0.093421	0.143492	0.006307	8.311369	0.011	0.093421256	0.142875893	0.006922643	8.275695606
0.012	0.101697	0.142876	0.006923	8.275696	0.012	0.101696952	0.142262653	0.007535884	8.240175331
0.013	0.109937	0.142263	0.007536	8.240175	0.013	0.109937127	0.141652045	0.008146492	8.204807513
0.014	0.118142	0.141652	0.008146	8.204808	0.014	0.118141935	0.141044057	0.00875448	8.169591498
0.015	0.126312	0.141044	0.008754	8.169591	0.015	0.126311526	0.140438679	0.009359858	8.134526635
0.016	0.134446	0.140439	0.00936	8.134527	0.016	0.134446053	0.1398359	0.009962637	8.099612274
0.017	0.142546	0.139836	0.009963	8.099612	0.017	0.142545665	0.139235707	0.01056283	8.06484777
0.018	0.150611	0.139236	0.010563	8.064848	0.018	0.150610513	0.138638091	0.011160446	8.030232479
0.019	0.158641	0.138638	0.01116	8.030232	0.019	0.158640745	0.13804304	0.011755497	7.995765761
0.02	0.166637	0.138043	0.011755	7.995766	0.02	0.166636511	0.137450542	0.012347994	7.961446978

Fig 2.48 (trace table of 8mm radius, 8.05gcm^{-3} ball in sunflower oil)

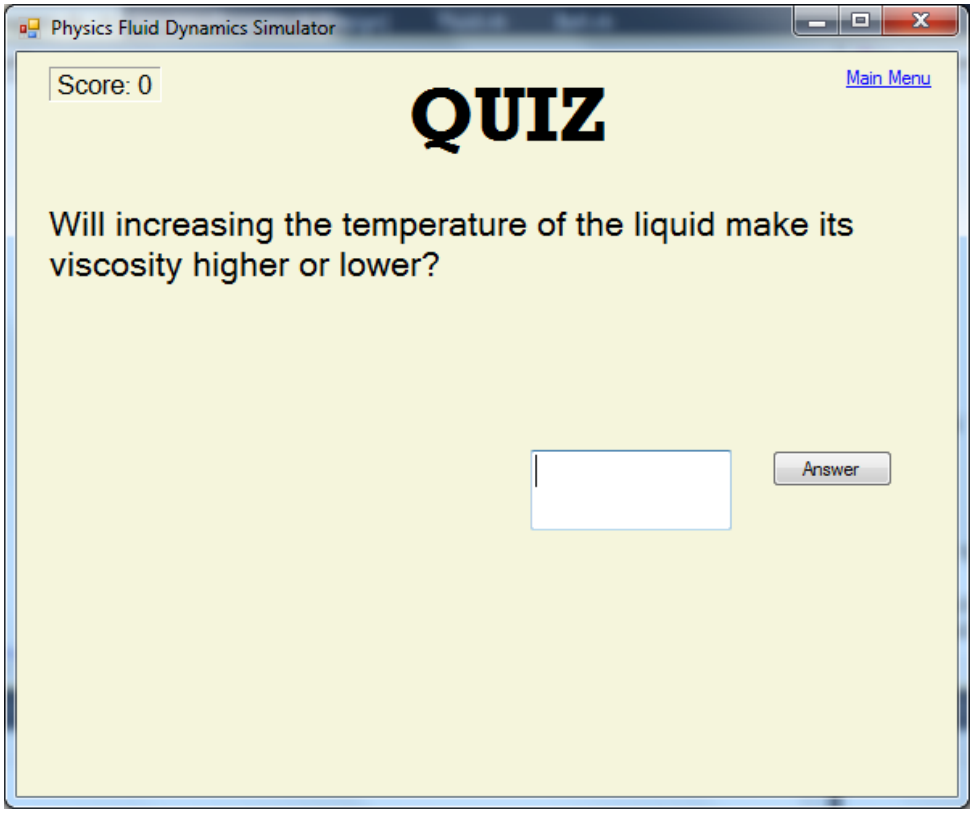


Fig 3.1 (shows a random starting question)

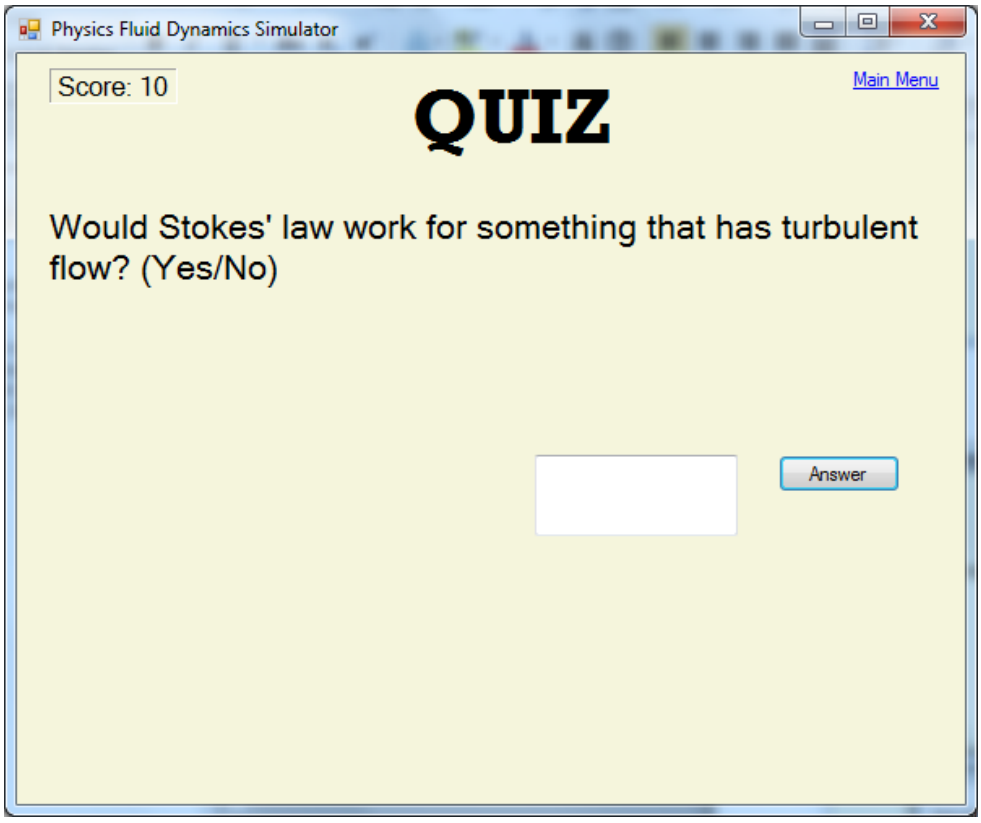


Fig 3.2 (shows another question selected randomly)

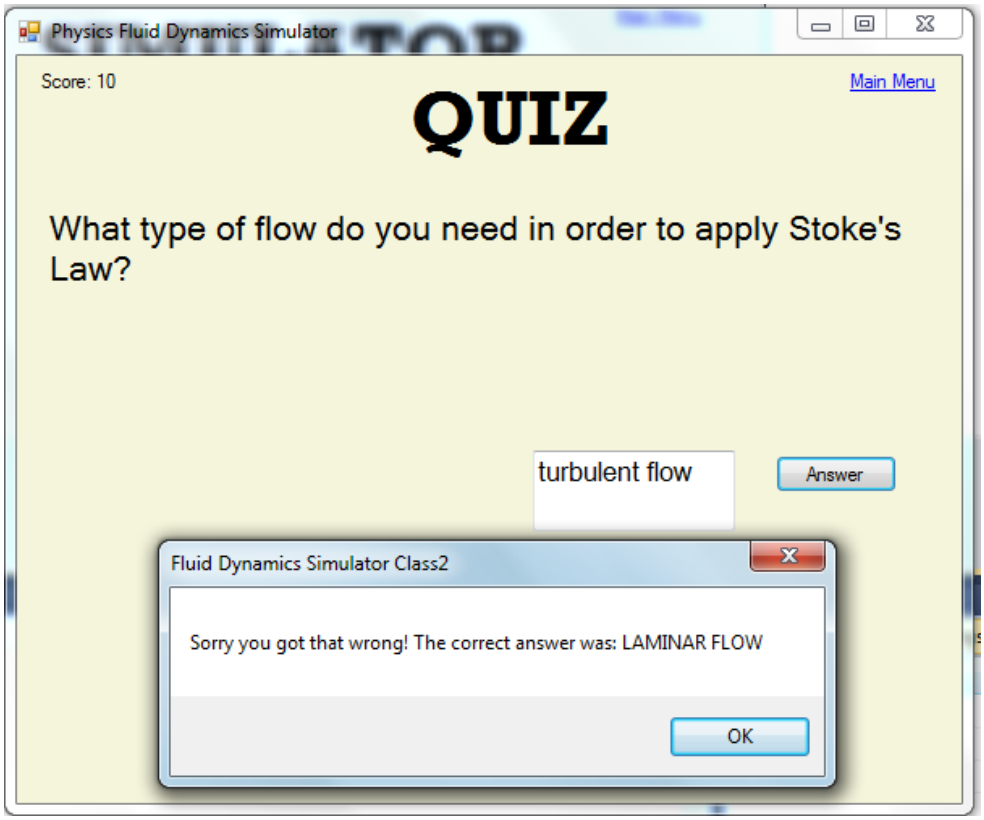


Fig 3.3 (shows the correct answer if the user got it wrong)

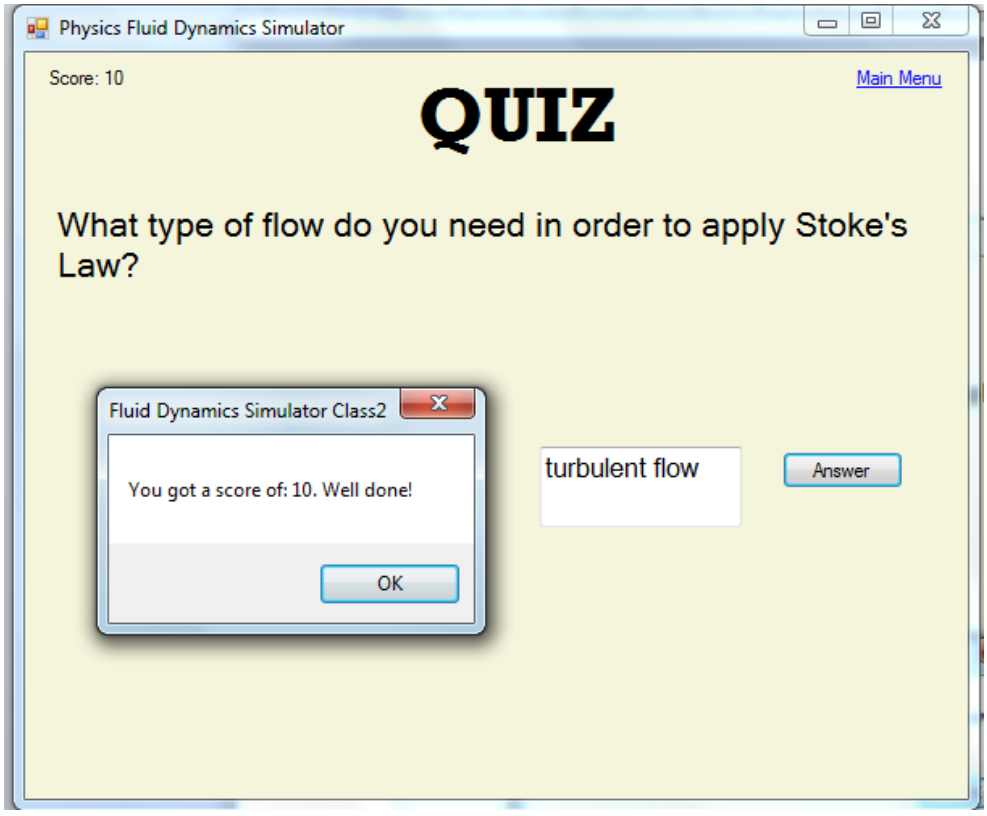


Fig 3.4 (shows the final score the user got)

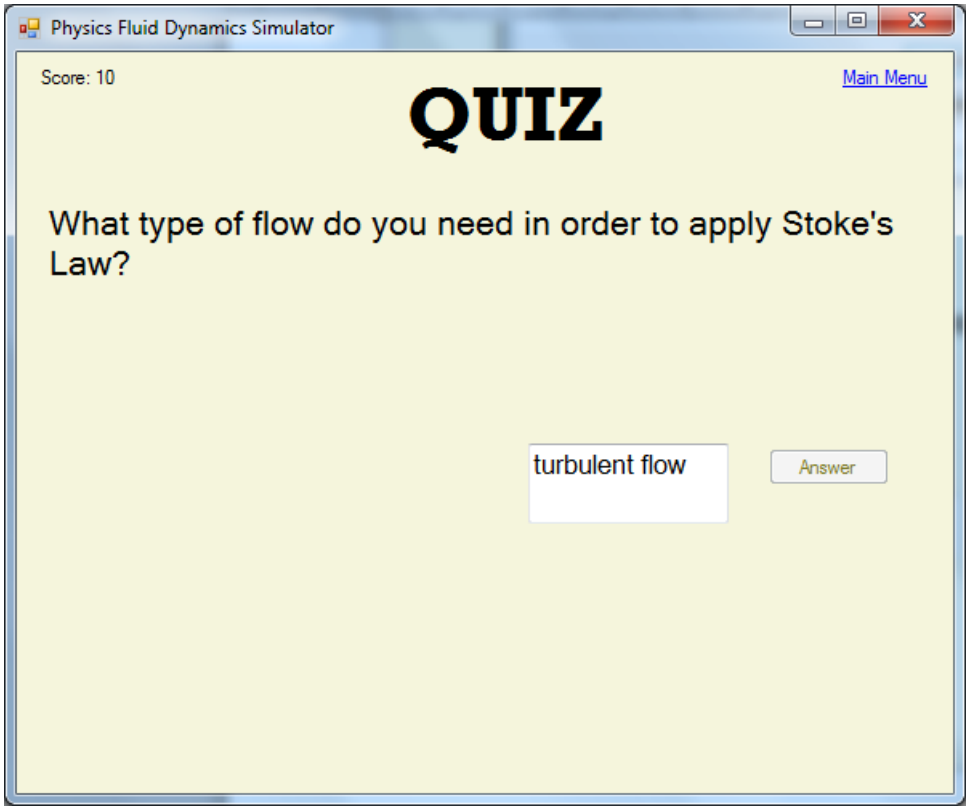


Fig 3.5 (disables the answer button to prevent the user from continuing the quiz)

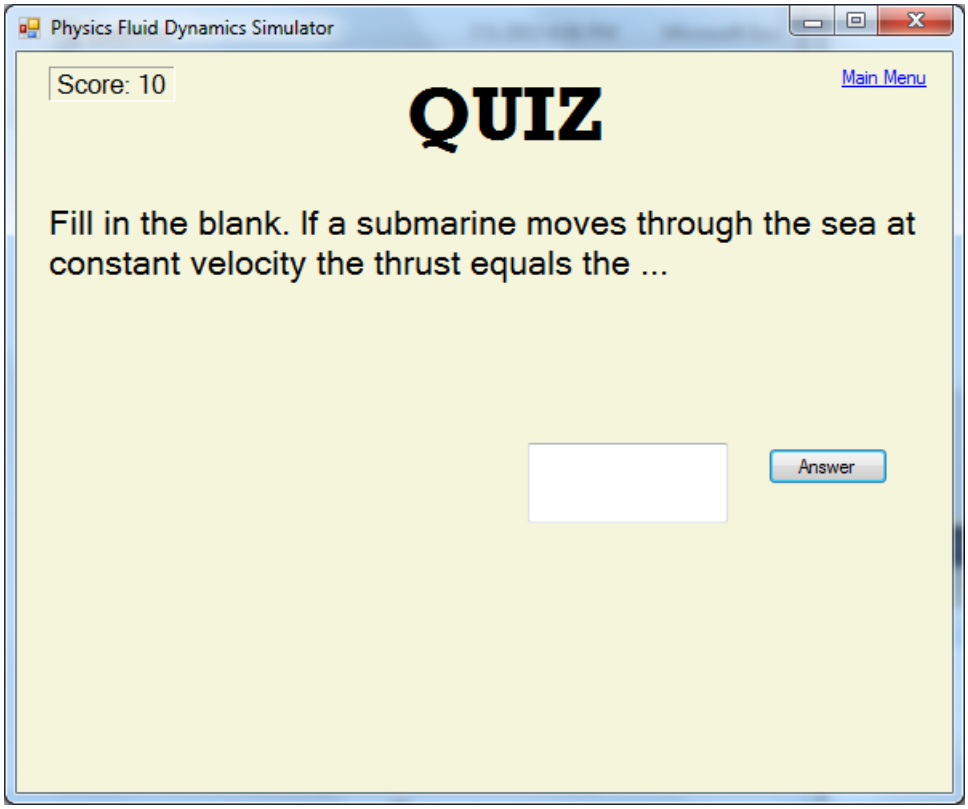


Fig 3.6 (displayed score on the top left hand corner shown to increase when a user gets a question right)

7.8 Code listing

7.8.1 Code for MainMenu form

```
Public Class MainMenu
    Dim audioDevice As New Microsoft.VisualBasic.Devices.Audio

    Private Sub Form1_FormClosed(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Me.FormClosed
        'Calls subroutine to close all forms
        Shutdown()
        End
    End Sub

    Private Sub Shutdown()
        'Closes all the forms that may be hidden
        SimulatorProgram.Close()
        QuickQuiz.Close()
        CalculationQuiz.Close()
    End Sub

    Private Sub Button1_Click(sender As System.Object, e As System.EventArgs) Handles
btnSimulator.Click
        'Opens up the SimulatorProgram form
        Dim Simulator As New SimulatorProgram
        Simulator.Show()
        'Closes this current form
        Me.Hide()
    End Sub

    Private Sub Button2_Click(sender As System.Object, e As System.EventArgs) Handles
btnQuickQuiz.Click
        'Opens up the QuickQuiz form
        Dim QuickQuiz As New QuickQuiz
        QuickQuiz.Show()
        'Closes the current form
        Me.Hide()
    End Sub

    Private Sub Button3_Click(sender As System.Object, e As System.EventArgs) Handles
btnCalculationQuiz.Click
        'Opens up the CalculationQuiz form
        Dim CalculationQuiz As New CalculationQuiz
        CalculationQuiz.Show()
        'Closes the current form
        Me.Hide()
    End Sub

    Private Sub button_clicked(ByVal sender As Object, ByVal e As EventArgs) Handles
btnSimulator.Click, btnQuickQuiz.Click, btnCalculationQuiz.Click
        'Plays a sound effect when a button is clicked
        audioDevice.Play(My.Resources.Click, AudioPlayMode.Background)
    End Sub

    Private Sub Form1_Load(sender As System.Object, e As System.EventArgs) Handles
MyBase.Load
        'Sets the backcolour of the form
        Me.BackColor = Color.Beige
    End Sub
End Class
```

7.8.2 Code for SimulatorProgram form

```
Imports System.IO

Public Class SimulatorProgram
    Dim myBall As New Ball
    Dim myFluid As New Fluid
    Dim mySimulator As New Simulator
    Dim Time As Single
    Dim data(5, 999) As String
    Dim position As Integer = 1

    Private Sub Form2_FormClosed(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Me.FormClosed
        'Closes the program
        Shutdown()
        End
    End Sub

    Private Sub Shutdown()
        'Closes all the other forms that is hidden
        MainMenu.Close()
        QuickQuiz.Close()
        CalculationQuiz.Close()
    End Sub

    Private Sub Form2_Load(sender As System.Object, e As System.EventArgs) Handles
MyBase.Load
        'Initiaise the initial components
        FormInit()
    End Sub

    Private Sub FormInit()
        'Loads the initial images such as the ball and tube
        myFluid.image = Tube
        myFluid.image.Image = My.Resources.Test_tube
        myball.image = Ball
        myBall.image.Image = My.Resources.Ball
        myball.Y = 210
        'Sets the initial properties of the ball
        LoadDefaultBall()
        'Loads colour background
        Me.BackColor = Color.Beige
    End Sub

    Private Sub LoadDefaultBall()
        'Sets the initial properties of the ball
        myball.radius = 2
        myball.density = 2
        lblRadius.Text = "Radius: " & scrRadius.Value & "mm"
        lblDensity.Text = "Density: " & scrDensity.Value & "g/cm3"
        myball.CalculateVolume(myball.radius)
        lblVolume.Text = "Volume: " & Math.Round(myball.volume, 10) & " m3"
        myball.Calculatemass(myball.volume, myball.density)
        lblWeight.Text = "Weight: " & Math.Round(myball.Getweight(), 5) & " N"
    End Sub

    Private Sub LinkLabel1_LinkClicked(sender As System.Object, e As
System.Windows.Forms.LinkLabelLinkClickedEventArgs) Handles LinkLabel1.LinkClicked
        'Takes them back to the main menu
        Dim MainMenu As New MainMenu
```

```
        MainMenu.Show()
        Me.Hide()
    End Sub

    Private Sub TrackBar1_Scroll(sender As System.Object, e As System.EventArgs)
Handles scrRadius.Scroll
        'Allows the user to change the radius of the ball
        'Changes the radius value of the ball
        myball.radius = scrRadius.Value
        'Adjusts the visual size of the ball
        changeSizeOfBall()
        'Resets all the variables of the simulation and enables the run button
        ReadyForNextRun()
        'Updates the output of the ball properties to the user
        ChangeSolidVariables()
    End Sub

    Private Sub TrackBar2_Scroll(sender As System.Object, e As System.EventArgs)
Handles scrDensity.Scroll
        'Allows the user to change the denisty of the ball
        'Changes the density of the ball
        myball.density = scrDensity.Value
        'Doesn't actually change size of the ball but needed to reset the ball to its
original position for next run
        changeSizeOfBall()
        'Resets all the variables of the simulation and enables the run button
        ReadyForNextRun()
        'Updates the output of the ball properties to the user
        ChangeSolidVariables()
        'Loads a different ball to show that this ball doesn't have the same density
as glass (marble ball) or steel
        myBall.image.Image = My.Resources.Ball
    End Sub

    Private Sub changeSizeOfBall()
        'Changes the size of the ball
        myball.image.Size = New Size(myball.radius * 10, myball.radius * 10)
        'Keeps location at the center of the tube
        myball.image.Location = New Point(253 - myball.radius * 5, 210 - myball.radius
* 5)
        'Sets the vertical position property of the ball
        myball.Y = 210 - myball.radius * 5
    End Sub

    Private Sub btnRun_Click(sender As System.Object, e As System.EventArgs) Handles
btnRun.Click
        'Runs the simulation
        'Checks if there is a fluid chosen
        If mySimulator.FluidChosen(myFluid.fluid) = True Then
            'Enables timer to start the animation
            Timer1.Enabled = True
            'Disables the radius and density slider to prevent the user from changing
the properties of the ball in the middle of the simulation
            scrRadius.Enabled = False
            scrDensity.Enabled = False
            'Disables the combo box for selecting the ball and fluid to prevent the
user from changing them in the middle of the simulation
            cboSelectBall.Enabled = False
            cboSelectFluid.Enabled = False
            'Hides the export button until simulation finishes
            btnExport.Visible = False
```

```
            'Resets the timer to 0
            Time = 0
        Else
            'Alerts the user that they haven't chosen a fluid yet
            MsgBox("No fluid is chosen!", MsgBoxStyle.Critical)
        End If
    End Sub

    Private Sub btnPause_Click(sender As System.Object, e As System.EventArgs) Handles
btnPause.Click
        'Pausese the simulation
        'Disables the timer
        Timer1.Enabled = False
    End Sub

    Private Sub Timer1_Tick(sender As System.Object, e As System.EventArgs) Handles
Timer1.Tick
        'This procedure controls everything that needs to be updated or checked every
timer tick
        'Updates all the properties of the simulation which keeps on changing
        UpdateVariables()
        'Controls the animation of the ball
        MoveBall()
        'Stops the ball when it is going out of the tube
        CheckGoingOutOfScreen()
    End Sub

    Private Sub UpdateVariables()
        'Updates the variables of the simulation
        'Stores the data of the variables in an array every timer tick to export the
data afterwards
        data(0, position) = Time
        data(1, position) = mySimulator.currentVelocity
        data(2, position) = mySimulator.Resultant
        data(3, position) = mySimulator.viscousDrag
        data(4, position) = mySimulator.Acceleration
        'Increments the position by one to start a new row for the next set of data
        position += 1
        'Outputs the time that has passed
        lblTime.Text = "Time: " & Math.Round(Time, 3)
        Time += 0.001
        'Calculates then outputs the upthrust
        mySimulator.CalculateUpthrust(myball.volume, myFluid.density)
        lblUpthrust.Text = "Upthrust: " & Math.Round(mySimulator.Upthrust, 5) & " N"
        'Calculates then outputs the viscous drag
        mySimulator.CalculateViscousDrag(myball.radius / 1000, myFluid.viscosity)
        lblViscousDrag.Text = "Viscous drag: " & Math.Round(mySimulator.viscousDrag, 5)
& " N"
        'Caclulates the acceleration (no need for output)
        mySimulator.CalculateAcceleration(myball.weight, myball.mass)
        'Calculates then outputs the current velocity
        mySimulator.CalculateNewVelocity()
        lblCurrentVelocity.Text = "Velocity: " &
Math.Round(mySimulator.currentVelocity, 5) & " ms-1"
        'Calculates then outputs the terminal velocity
        mySimulator.CalculateTerminalVeloctiy(myball.radius, myball.density,
myFluid.density, myFluid.viscosity)
        lblTerminalVelocity.Text = "Terminal velocity: " &
Math.Round(mySimulator.TerminalVelocity, 5) & " ms-1"
    End Sub
```

```
    Private Sub MoveBall()
        'Sets the myball.image.top property to the myball.Y property which is getting
updated
        myball.image.Top = myball.Y
        'Calculates the distance it should move
        mySimulator.CalculateDistance()
        'Moves the position of the ball by a set distance
        myball.Y += mySimulator.Distance
    End Sub

    Private Sub CheckGoingOutOfScreen()
        'Makes sure the ball doesn't go out of the tube
        If myball.Y + myball.image.Height > 438 Then
            'Allows the user to set up the next run
            PrepareForNextRun()
        End If
    End Sub

    Private Sub ChangeSolidVariables()
        'Changes the output of the variables of the ball
        'Outputs the new values for the radius and density
        lblRadius.Text = "Radius: " & scrRadius.Value & "mm"
        lblDensity.Text = "Density: " & scrDensity.Value & "g/cm3"
        'Calculates the new volume and outputs the new volume
        myball.CalculateVolume(myball.radius)
        lblVolume.Text = "Volume: " & Math.Round(myball.volume, 10) & " m3"
        'Calculates the new mass and outputs the new mass
        myball.Calculatemass(myball.volume, myball.density)
        lblWeight.Text = "Weight: " & Math.Round(myball.Getweight(), 5) & " N"
    End Sub

    Private Sub PrepareForNextRun()
        'disables the timer and run button
        Timer1.Enabled = False
        btnRun.Enabled = False
        'Enables the user to set a new radius ad density
        scrRadius.Enabled = True
        scrDensity.Enabled = True
        cboSelectBall.Enabled = True
        cboSelectFluid.Enabled = True
        'Gives the user an option to export data
        btnExport.Visible = True
    End Sub

    Private Sub cboSelectBall_SelectedIndexChanged(sender As System.Object, e As
System.EventArgs) Handles cboSelectBall.SelectedIndexChanged
        'Prepares for next run since new ball is selected
        ReadyForNextRun()
        'Changes the material therefore density, radius and picture of the ball
        SelectBall()
        'Outputs the new properties of the ball
        ChangeSolidVariables()
    End Sub

    Private Sub ReadyForNextRun()
        'Enables the run button for next run
        btnRun.Enabled = True
        'Resets all the variables for next run
        mySimulator.currentVelocity = 0
        mySimulator.TerminalVelocity = 0
        mySimulator.viscousDrag = 0
```

```
        mySimulator.Resultant = 0
        mySimulator.Acceleration = 0
        mySimulator.oldVelocity = 0
        'Resets the data stored in the data array to be exported to a csv file
        For x = 1 To Time * 1000 + 1
            data(0, x) = ""
            data(1, x) = ""
            data(2, x) = ""
            data(3, x) = ""
            data(4, x) = ""
        Next
        'Resets the position of the data array which the program is writing to back to
one
        position = 1
    End Sub

    Private Sub SelectBall()
        'Select case statement to assign the ball its properties according to what was
selected
        Select Case cboSelectBall.Text
            Case "8mm Steel Ball"
                myball.Material = "steel"
                myball.radius = 8
                myball.density = myball.GetDensity()
                myBall.image.Image = My.Resources.Steel_ball
            Case "6mm Steel Ball"
                myball.Material = "steel"
                myball.radius = 6
                myball.density = myball.GetDensity()
                myBall.image.Image = My.Resources.Steel_ball
            Case "6mm Marble Ball"
                myball.Material = "marble"
                myball.radius = 6
                myball.density = myball.GetDensity()
                myBall.image.Image = My.Resources.Marble_ball
            Case "8mm Marble Ball"
                myball.Material = "marble"
                myball.radius = 8
                myball.density = myball.GetDensity()
                myBall.image.Image = My.Resources.Marble_ball
        End Select
        'Changes the size of the picture of the ball so that the user can see the
difference
        changeSizeOfBall()
    End Sub

    Private Sub cboSelectFluid_SelectedIndexChanged(sender As System.Object, e As
System.EventArgs) Handles cboSelectFluid.SelectedIndexChanged
        'Assigns what fluid is being used to the fluid property in the fluid class
        SelectFluid()
        'Outputs the new fluid properties
        ChangeFluidVariables()
    End Sub

    Private Sub SelectFluid()
        'Changes the value of fluid in the fluid class based on what was selected
        Select Case cboSelectFluid.Text
            Case "Glycerol"
                myFluid.fluid = "Glycerol"
            Case "Sunflower oil"
                myFluid.fluid = "Sunflower oil"
```

```
            Case "Coconut oil"
                myFluid.fluid = "Coconut oil"
            Case "Olive oil"
                myFluid.fluid = "Olive oil"
            Case Else
                myFluid.fluid = ""
        End Select
        'Changes the properties of the fluid
        myFluid.density = myFluid.GetDensity()
        myFluid.viscosity = myFluid.GetViscosity()
        'Changes the image of the fluid
        myFluid.Changecolour()
        'Changes the backcolour of the ball to match that of the fluid so it looks
transparent
        myball.ChangeColour(myFluid.fluid)
        'Gets ready for next run by reseting all the variables
        ReadyForNextRun()
        'Puts ball back to initial position
        changeSizeOfBall()
    End Sub

    Private Sub ChangeFluidVariables()
        'Outputs the new values for density and viscosity
        lblFluidDensity.Text = "Fluid density: " & myFluid.density & " g/cm3"
        lblViscosity.Text = "Viscosity: " & myFluid.viscosity & " Pas"
    End Sub

    Private Sub btnExport_Click(sender As System.Object, e As System.EventArgs)
Handles btnExport.Click
        'Creates a streamwrite class
        Dim StreamToWrite As StreamWriter
        Dim FileLocation As String

        'User inputs where they want to write the file and the file name
        FileLocation = InputBox("Please type in the file path followed by the file
name which you want it to be called followed by .csv")
        If FileLocation <> "" Then
            Try
                StreamToWrite = New StreamWriter(FileLocation)
                'Write the first row of the table
                StreamToWrite.Write("Time (s)" & "," & "Velocity (m/s)" & "," &
"Resultant force (N)" & "," & "Viscous drag (N)" & "," & "Acceleration (m/s^2)")
                'Writes all the information that doesn't need to be updated every
timer tick
                StreamToWrite.Write(",," & "Terminal velocity" & "," &
mySimulator.TerminalVelocity & "," & "m/s" & "," & "Upthrust" & "," & _
                                    mySimulator.Upthrust & "," & "N" & "," & "Radius"
& "," & myBall.radius & "," & "mm" & "," & "Weight" & "," & _
                                    myBall.weight & "," & "N" & "," & "Volume" & "," &
myBall.volume & "," & "m^3" & "," & "Density of ball" & "," & _
                                    myBall.density & "," & "g/cm^3" & "," & "Fluid" &
"," & myFluid.fluid & "," & "Fluid density" & "," & myFluid.density & _
                                    "," & "g/cm^3" & "," & "Viscosity" & "," &
myFluid.viscosity & "," & "Pas")
                'Creates next row
                StreamToWrite.Write(vbCrLf)
                'Writes the data row by row from the array that stored the values of
simulation just now
                'The time is multiplied by 1000 because they go in 0.001 second
intervals so 0.001*1000 = 1 and + 1 because time starts from 0
                For x = 1 To Time * 1000 + 1
```

```
                    StreamToWrite.Write(data(0, x) & ",")
                    StreamToWrite.Write(data(1, x) & ",")
                    StreamToWrite.Write(data(2, x) & ",")
                    StreamToWrite.Write(data(3, x) & ",")
                    StreamToWrite.Write(data(4, x))
                    StreamToWrite.Write(vbCrLf)
                Next
                'Notifies user data has been exported
                MsgBox("Data exported!", MsgBoxStyle.Exclamation)
                'Closes streamwriter file so can write again next time
                StreamToWrite.Close()
            Catch ex As Exception
                'Alerts the user that an error has occured
                MsgBox("Invalid file location/File is open", MsgBoxStyle.Critical)
            End Try
        End If
    End Sub
End Class
```

7.8.3 Code for QuickQuiz form

```
Imports System.IO

Public Class QuickQuiz
    Dim myQuiz As New Quiz

    Private Sub Form3_FormClosed(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Me.FormClosed
        'Closes the program
        Shutdown()
        End
    End Sub

    Private Sub Shutdown()
        'Closes the hidden forms
        MainMenu.Close()
        SimulatorProgram.Close()
        CalculationQuiz.Close()
    End Sub

    Private Sub Form3_Load(sender As System.Object, e As System.EventArgs) Handles
MyBase.Load
        'Uses randomize to make sure the order of questions are different each time
        Randomize()
        'Tells the quiz class that this is the QuickQuiz not the CalculationQuiz and
so reads in the correct file for the questions
        myQuiz.readFile("quick")
        'Sets the backcolour of the form
        Me.BackColor = Color.Beige
        'Calls procedure to display question
        OutputQuestion()
    End Sub

    Private Sub OutputQuestion()
        'Displays the question on the label with the randomly selected question
        lblQuestion.Text = myQuiz.GenerateQuestion()
    End Sub

    Private Sub LinkLabel1_LinkClicked(sender As System.Object, e As
System.Windows.Forms.LinkLabelLinkClickedEventArgs) Handles LinkLabel1.LinkClicked
        'Displays main menu
```

```
        Dim MainMenu As New MainMenu
        MainMenu.Show()
        'Hides the current form
        Me.Hide()
    End Sub

    Private Sub btnAnswer_Click(sender As System.Object, e As System.EventArgs)
Handles btnAnswer.Click
        Dim Correct As Boolean = True
        'Checks the answer inputted by the user
        myQuiz.CheckAnswer(txtAnswer.Text, Correct)
        'Updates score
        lblScore.Text = "Score: " & myQuiz.Score
        'Clears textbox space for user's answer and prepares next question if the user
got it rihgt.
        'Otherwise, disables the answer button meaning the quiz has finished
        If Correct = True Then
            txtAnswer.Text = ""
            Proceed()
        Else
            btnAnswer.Enabled = False
        End If
    End Sub

    Private Sub Proceed()
        'Updates array so same question won't be displayed twice
        myQuiz.UpdateArray()
        'Outputs new question
        OutputQuestion()
    End Sub
End Class
```

7.8.4 Code for CalculationQuiz form

```
Imports System.IO

Public Class CalculationQuiz
    Dim myQuiz As New Quiz

    Private Sub Form3_FormClosed(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Me.FormClosed
        'Closes the program
        Shutdown()
        End
    End Sub

    Private Sub Shutdown()
        'Closes the hidden forms
        MainMenu.Close()
        SimulatorProgram.Close()
        QuickQuiz.Close()
    End Sub

    Private Sub Form3_Load(sender As System.Object, e As System.EventArgs) Handles
MyBase.Load
        'Uses randomize to make sure the order of questions are different each time
        Randomize()
        'Tells the quiz class that this is the CalculationQuiz not the QuickQuiz and
so reads in the correct file for the questions
        myQuiz.readFile("calculate")
        'Sets the backcolour of the form
```

```
        Me.BackColor = Color.Beige
        'Calls procedure to display question
        OutputQuestion()
    End Sub

    Private Sub OutputQuestion()
        'Displays the question on the label with the randomly selected question
        lblQuestion.Text = myQuiz.GenerateQuestion()
    End Sub

    Private Sub LinkLabel1_LinkClicked(sender As System.Object, e As
System.Windows.Forms.LinkLabelLinkClickedEventArgs) Handles LinkLabel1.LinkClicked
        'Displays main menu
        Dim MainMenu As New MainMenu
        MainMenu.Show()
        'Hides the current form
        Me.Hide()
    End Sub

    Private Sub btnAnswer_Click(sender As System.Object, e As System.EventArgs)
Handles btnAnswer.Click
        Dim Correct As Boolean = True
        'Checks the answer inputted by the user
        myQuiz.CheckAnswer(txtAnswer.Text, Correct)
        'Updates score
        lblScore.Text = "Score: " & myQuiz.Score
        'Clears textbox space for user's answer and prepares next question if the user
got it rihgt.
        'Otherwise, disables the answer button meaning the quiz has finished
        If Correct = True Then
            txtAnswer.Text = ""
            Proceed()
        Else
            btnAnswer.Enabled = False
        End If
    End Sub

    Private Sub Proceed()
        'Updates array so same question won't be displayed twice
        myQuiz.UpdateArray()
        'Outputs new question
        OutputQuestion()
    End Sub
End Class
```

7.8.5 Code for Simulator class

```
Public Class Simulator
    Private p_OldVelocity As Double = 0
    Private p_CurrentVelocity As Double
    Private p_TerminalVelocity As Double
    Private p_ViscousDrag As Double
    Private p_Upthrust As Double
    Private p_Resultant As Double
    Private p_Acceleration As Double
    Private p_Distance As Double
    Const g = 9.81
    Const time = 0.001

    Public Sub CalculateUpthrust(volume, densityOfFluid)
        'Caclulates up-thrust using volume and density
```

```
        'Multiplied by 1000 to get it in SI units
        p_Upthrust = volume * densityOfFluid * 1000 * g
    End Sub

    Public Sub CalculateAcceleration(Weight, Mass)
        'Calculates the resultant force
        p_Resultant = Weight - (p_ViscousDrag + p_Upthrust)
        'Uses Newton's 2nd law to get the acceleration
        p_Acceleration = p_Resultant / Mass
    End Sub

    Public Sub CalculateNewVelocity()
        'Uses the SUVAT equation v = u + at
        p_CurrentVelocity = p_OldVelocity + p_Acceleration * time
        'Updates the old velocity to become the new velocity
        p_OldVelocity = p_CurrentVelocity
    End Sub

    Public Sub CalculateDistance()
        'Calculates the distance it should move by using distance = speed * time
        'Multiplies the time by 100000 to get noticable movement because time in order
of magnitude 10^-3 and velocities are very small
        p_Distance = p_CurrentVelocity * time * 100000
    End Sub

    Public Sub CalculateTerminalVeloctiy(radius, solidDensity, fluidDensity, viscosity)
        'Calculates the terminal velocity using equation (2r^2(ρs – ρf))/(9η)
        'Divides and multiplies by 1000 to get it into SI units
        p_TerminalVelocity = (2 * (radius / 1000) ^ 2 * g * (solidDensity * 1000 -
fluidDensity * 1000)) / (9 * viscosity)
    End Sub

    Public Sub CalculateViscousDrag(radius, viscosity)
        'Calculates the viscous drag acting on the ball using Stokes' law
        p_ViscousDrag = 6 * System.Math.PI * p_CurrentVelocity * radius * viscosity
    End Sub

    Public Function FluidChosen(fluid) As Boolean
        'Checks if there is fluid chosen and returns true if there is or false if
there isn't
        FluidChosen = True
        If fluid = "" Then
            FluidChosen = False
        End If
        Return FluidChosen
    End Function

    Property currentVelocity
        Get
            Return p_CurrentVelocity
        End Get
        Set(value)
            p_CurrentVelocity = value
        End Set
    End Property

    Property oldVelocity
        Get
            Return p_OldVelocity
        End Get
        Set(value)
```

```
            p_OldVelocity = value
        End Set
    End Property

    Property viscousDrag
        Get
            Return p_ViscousDrag
        End Get
        Set(value)
            p_ViscousDrag = value
        End Set
    End Property

    Property Upthrust
        Get
            Return p_Upthrust
        End Get
        Set(value)
            p_Upthrust = value
        End Set
    End Property

    Property Resultant
        Get
            Return p_Resultant
        End Get
        Set(value)
            p_Resultant = value
        End Set
    End Property

    Property TerminalVelocity
        Get
            Return p_TerminalVelocity
        End Get
        Set(value)
            p_TerminalVelocity = value
        End Set
    End Property

    Property Acceleration
        Get
            Return p_Acceleration
        End Get
        Set(value)
            p_Acceleration = value
        End Set
    End Property

    Property Distance
        Get
            Return p_Distance
        End Get
        Set(value)
            p_Distance = value
        End Set
    End Property
End Class
```

7.8.6 Code for Ball class

```
Public Class Ball
    Private p_radius As Integer
    Private p_density As Double
    Private p_top As Double
    Private p_material As String
    Private p_volume As Double
    Private p_mass As Double
    Private p_weight As Double
    Private p_image As PictureBox
    Const g = 9.81
    Const steelDensity = 8.05
    Const marbleDensity = 2.4

    Public Sub ChangeColour(fluid)
        'Changes the backcolour of the ball according to the fluid so that the
backcolour of the ball matched the fluid colour
        Select Case fluid
            Case "Glycerol"
                p_image.BackColor = Color.FromArgb(255, 215, 215, 215)
            Case "Sunflower oil"
                p_image.BackColor = Color.Yellow
            Case "Coconut oil"
                p_image.BackColor = Color.FromArgb(255, 225, 230, 30)
            Case "Olive oil"
                p_image.BackColor = Color.FromArgb(255, 200, 200, 60)
            Case Else
                p_image.BackColor = Control.DefaultBackColor
        End Select
    End Sub

    Public Sub CalculateVolume(radius As Integer)
        'Radius is in units of mm
        p_volume = (4 * System.Math.PI * (radius / 1000) ^ 3) / 3
    End Sub

    Public Sub Calculatemass(volume, density)
        'density is in g/cm^3 so have to convert it to kg/m^3 by multiplying by 1000
        p_mass = volume * density * 1000
    End Sub

    Public Function Getweight()
        'Calculates the weight using mass
        p_weight = p_mass * g
        Return p_weight
    End Function

    Public Function GetDensity()
        Dim density As Double
        'Assigns the density according to what material and returns the density
        If p_material = "steel" Then
            density = steelDensity
        ElseIf p_material = "marble" Then
            density = marbleDensity
        End If
        Return density
    End Function

    Property image As PictureBox
        Get
```

```
            Return p_image
        End Get
        Set(value As PictureBox)
            p_image = value
        End Set
    End Property

    Property Material As String
        Get
            Return p_material
        End Get
        Set(value As String)
            p_material = value
        End Set
    End Property

    Property Y As Double
        Get
            Return p_top
        End Get
        Set(value As Double)
            p_top = value
        End Set
    End Property

    Property radius As Integer
        Get
            Return p_radius
        End Get
        Set(value As Integer)
            p_radius = value
        End Set
    End Property

    Property density As Double
        Get
            Return p_density
        End Get
        Set(value As Double)
            p_density = value
        End Set
    End Property

    Property volume As Double
        Get
            Return p_volume
        End Get
        Set(value As Double)
            p_volume = value
        End Set
    End Property

    Property weight
        Get
            Return p_weight
        End Get
        Set(value)
            p_weight = value
        End Set
    End Property
```

```
    Property mass
        Get
            Return p_mass
        End Get
        Set(value)
            p_mass = value
        End Set
    End Property
End Class
```

7.8.7 Code for Fluid class

```
Public Class Fluid
    Private p_fluid As String
    Private p_density As Double
    Private p_viscosity As Double
    Private p_image As PictureBox
    Const GlycerolDensity = 1.26
    Const SunFlowerOilDensity = 0.93
    Const CoconutOilDensity = 0.924
    Const OliveOilDensity = 0.86
    Const GlycerolViscosity = 1.412
    Const SunFlowerOilViscosity = 0.4914
    Const CoconutOilViscosity = 0.06
    Const OliveOilViscosity = 0.1075

    Public Sub Changecolour()
        'Loads a tube with a different colour depending on what fluid is selected
        Select Case p_fluid
            Case "Glycerol"
                p_image.Image = My.Resources.Glycerol_test_tube
            Case "Sunflower oil"
                p_image.Image = My.Resources.Sunflower_test_tube
            Case "Coconut oil"
                p_image.Image = My.Resources.Coconut_oil_test_tube
            Case "Olive oil"
                p_image.Image = My.Resources.Olive_oil_test_tube
            Case Else
                p_image.Image = My.Resources.Test_tube
        End Select
    End Sub

    Public Function GetDensity()
        'Gets the density of the fluid and returns it according to what is the
selected fluid
        Select Case p_fluid
            Case "Glycerol"
                p_density = GlycerolDensity
            Case "Sunflower oil"
                p_density = SunFlowerOilDensity
            Case "Coconut oil"
                p_density = CoconutOilDensity
            Case "Olive oil"
                p_density = OliveOilDensity
        End Select
        Return p_density
    End Function

    Public Function GetViscosity()
        'Gets viscosity of the fluid and returns it according to what is the selected
fluid
```

```
        Select Case p_fluid
            Case "Glycerol"
                p_viscosity = GlycerolViscosity
            Case "Sunflower oil"
                p_viscosity = SunFlowerOilViscosity
            Case "Coconut oil"
                p_viscosity = CoconutOilViscosity
            Case "Olive oil"
                p_viscosity = OliveOilViscosity
        End Select
        Return p_viscosity
    End Function

    Property image As PictureBox
        Get
            Return p_image
        End Get
        Set(value As PictureBox)
            p_image = value
        End Set
    End Property

    Property fluid As String
        Get
            Return p_fluid
        End Get
        Set(value As String)
            p_fluid = value
        End Set
    End Property

    Property density As Single
        Get
            Return p_density
        End Get
        Set(value As Single)
            p_density = value
        End Set
    End Property

    Property viscosity As Single
        Get
            Return p_viscosity
        End Get
        Set(value As Single)
            p_viscosity = value
        End Set
    End Property
End Class
```

7.8.8 Code for Quiz class

```
Imports System.IO

Public Class Quiz
    Private p_score As Integer
    Private z As Integer
    Private x As Integer = 0
    Private Questions(50) As String
    Private questionNo As Integer
    Private audioDevice As New Microsoft.VisualBasic.Devices.Audio
```

```
    Public Sub ReadFile(quiztype)
        'Creates streamreader class
        Dim fileReader As StreamReader
        'Reads the short questions for QuickQuiz by default
        fileReader = New
StreamReader(My.Computer.FileSystem.SpecialDirectories.MyDocuments & "\Short
questions.csv")
        'If it is the CalculationQuiz thenr reads in the calculation questions
        If quiztype = "calculate" Then
            fileReader = New
StreamReader(My.Computer.FileSystem.SpecialDirectories.MyDocuments & "\Long
questions.csv")
        End If
        'Stores each line of the file in an array until end of file is reached
        Do
            Questions(x) = fileReader.ReadLine()
            x += 1
        Loop Until fileReader.EndOfStream
        'Closes filereader so it can read files again later
        fileReader.Close()
        'Redims the Questions array which was initially Questions(50) to how many
questions there actually are on the document to save memory space
        ReDim Preserve Questions(x)
    End Sub

    Public Sub CheckAnswer(ByVal Answer As String, ByRef Correct As Boolean)
        'Converts the use input to capital letters so that it is not case sensitive
        'Checks user's answer with answer part of the line
        If UCase(Answer) = Mid(Questions(questionNo), z + 2, Len(Questions(questionNo))
- z + 2) Then
            'Executes these lines if the user got it right
            'Plays sound effect indicating the user got it right
            audioDevice.Play(My.Resources.Correct, AudioPlayMode.Background)
            'Updates score
            p_score += 10
        Else
            'Execute these lines if the user got it wrong
            'Plays a sound effect indicating the user got it wrong
            audioDevice.Play(My.Resources.Incorrect, AudioPlayMode.Background)
            Correct = False
            'Outputs the correct answer for user's knowledge
            MsgBox("Sorry you got that wrong! The correct answer was: " &
Mid(Questions(questionNo), z + 2, Len(Questions(questionNo)) - z + 2))
            'Outputs their score and a message depending on how high their score is
            If Score > 50 Then
                MsgBox("You got a score of: " & p_score & ". Well done!")
            ElseIf Score > 0 Then
                MsgBox("You got a score of: " & p_score & ". Good effort!")
            Else
                MsgBox("You got a score of: " & p_score & ". Need to work harder!")
            End If
        End If
    End Sub

    Public Sub UpdateArray()
        Dim y As Integer
        'Shifts the position of questions in the array below the selected question up
by a position of 1
        For y = questionNo To (x - 1)
            Questions(y) = Questions(y + 1)
```

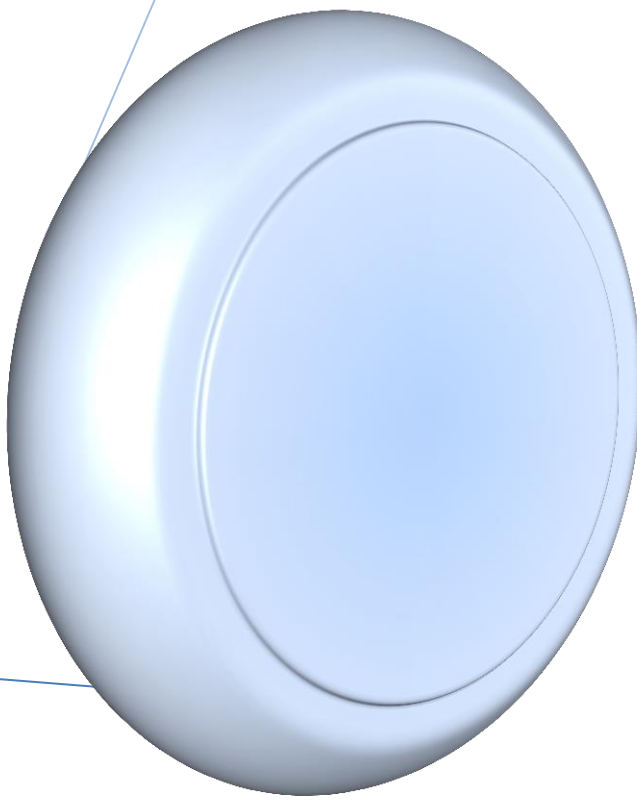
```
        Next
        'Subtract 1 from the value of x to show how many questions are left remaining
        x -= 1
        'Redim preserves the array so that a 'blank' question would not be picked
        ReDim Preserve Questions(x)
    End Sub

    Function GenerateQuestion()
        Dim firstpos As Boolean = False
        Dim secondpos As Boolean = False
        Dim y As Integer
        'x = 0 when all questions answered
        If x = 0 Then
            MsgBox("Well done you finished the game!")
            MsgBox("You got a score of: " & p_score & ". Well done!")
            Return Nothing
        Else
            y = 1
            'Chooses a random question
            questionNo = CInt(Int(Rnd() * x))
            'Gets question part of the csv file
            Do
                If Mid(Questions(questionNo), y, 1) = "," Then
                    firstpos = True
                Else
                    y += 1
                End If
            Loop Until firstpos = True
            z = y + 2
            Do
                If Mid(Questions(questionNo), z, 1) = "," Then
                    secondpos = True
                Else
                    z += 1
                End If
            Loop Until secondpos = True
            'Outputs the question
            Return Mid(Questions(questionNo), y + 2, z - (y + 2))
        End If
    End Function

    Property Score As Integer
        Get
            Return p_score
        End Get
        Set(value As Integer)
            p_score = value
        End Set
    End Property
End Class
```

User manual

For Tanglin Trust School Physics students and teachers

This is a step by step guide on how to use the physics fluid dynamics simulator program and tells you what each object on the form does. It also goes through some troubleshooting tips.

Victor Sim
3/26/2015



User manual

Contents

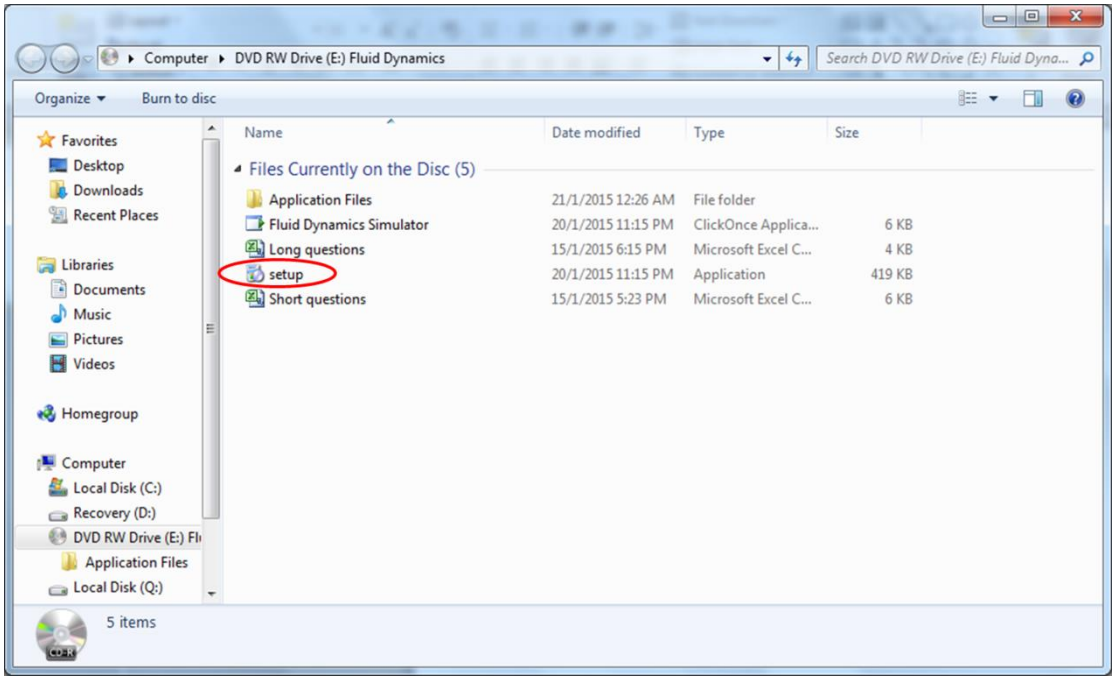
1. Introduction

This application is designed for the use of Tanglin Trust School students who are taking the Edexcel AS Physics exam. The application is focused on the fluid dynamics section of the course. There are 3 modes available in the program. In the simulator mode, students may experiment how changing the radius and density of the ball affects the terminal velocity of the ball as well as other variables. They are also able to change the fluid therefore changing the viscosity and density of the fluid as well to see how these factors play a part in the forces acting on the ball and how it affects the terminal velocity. If the students want, they can also export the data in the simulation as a .csv file to keep as a record and they are able to observe how the values of the variables are changing in their own time. The other 2 modes are quiz modes, one being a quiz based on quick answers and the other is based on answers requiring calculations. It will keep outputting new questions if the user gets it correct until the user gets it wrong or the user has finished answering all questions available. If the user inputs the wrong answer, the right answer will be displayed to them.

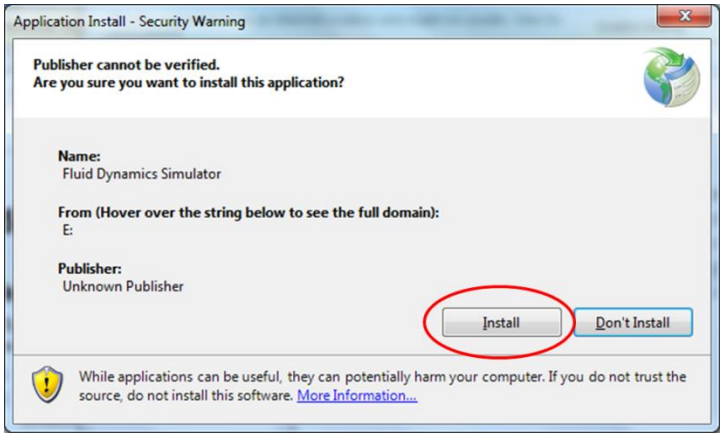
2. Installing the system

The system itself is relatively easy to install for first time usage. To install the system you must do the following:

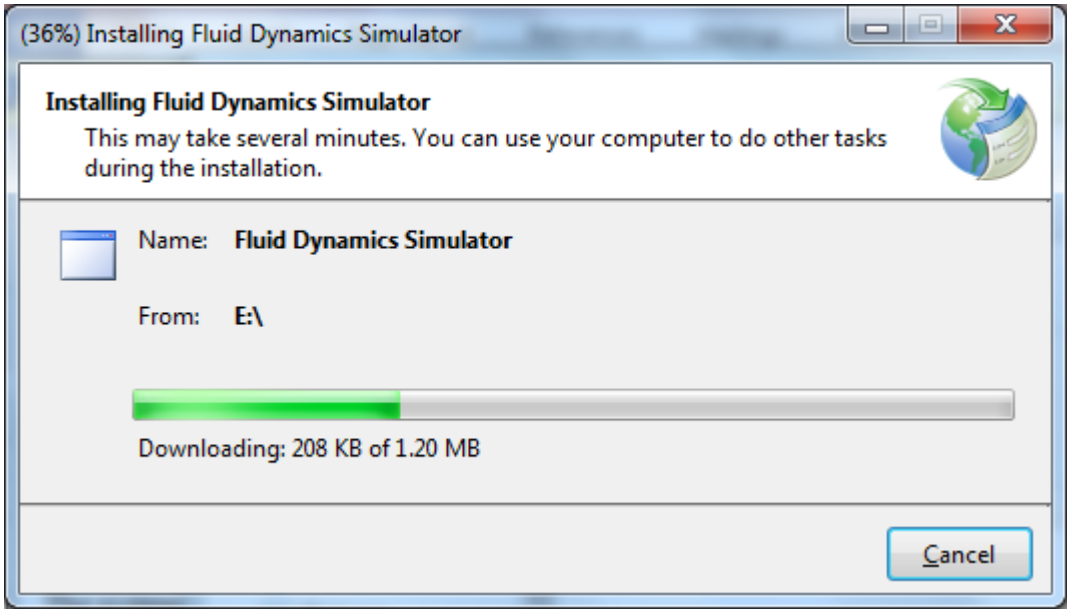
1. First insert the CD with the system on it into the CD-drive of your computer
2. Double click on the setup icon



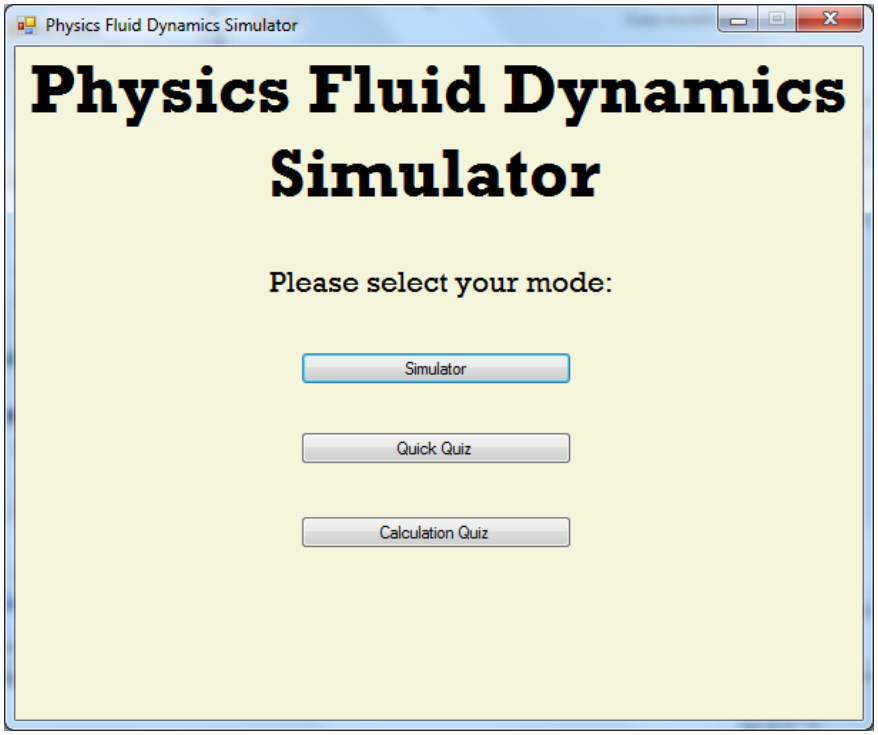
3. Then click on the install button



4. You may see this screen during the installation process

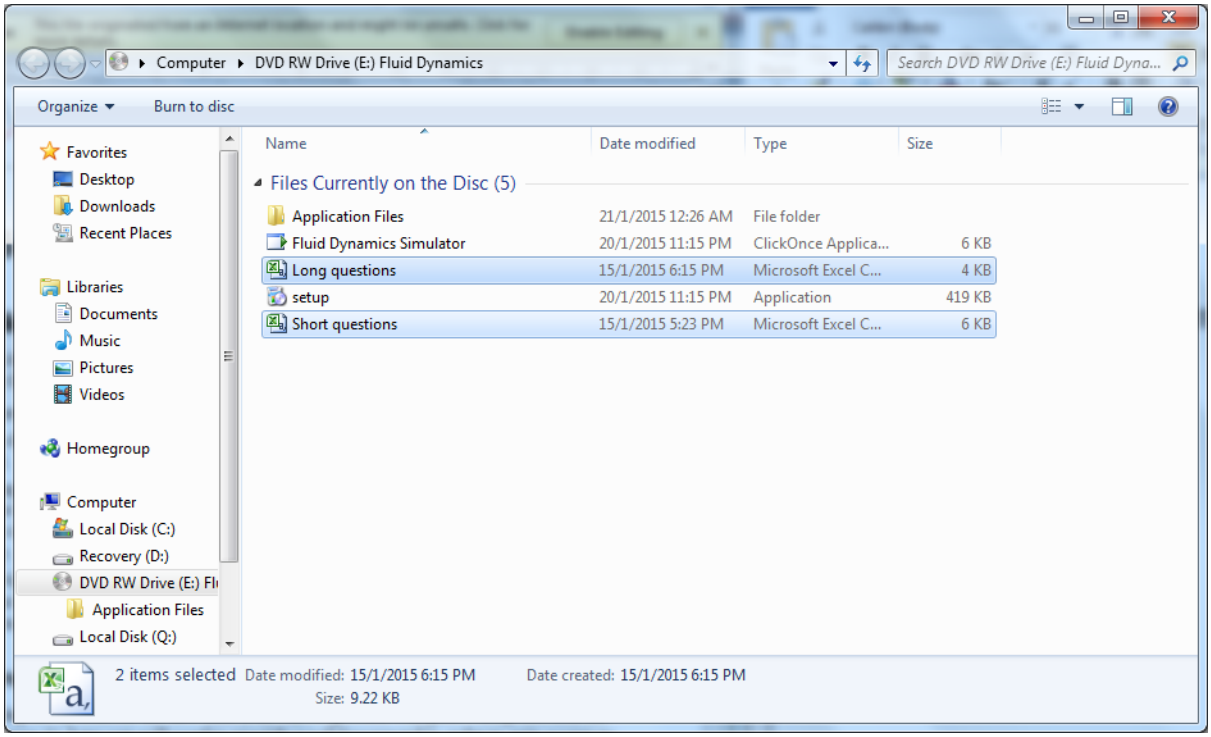


5. After that the application program should be loaded

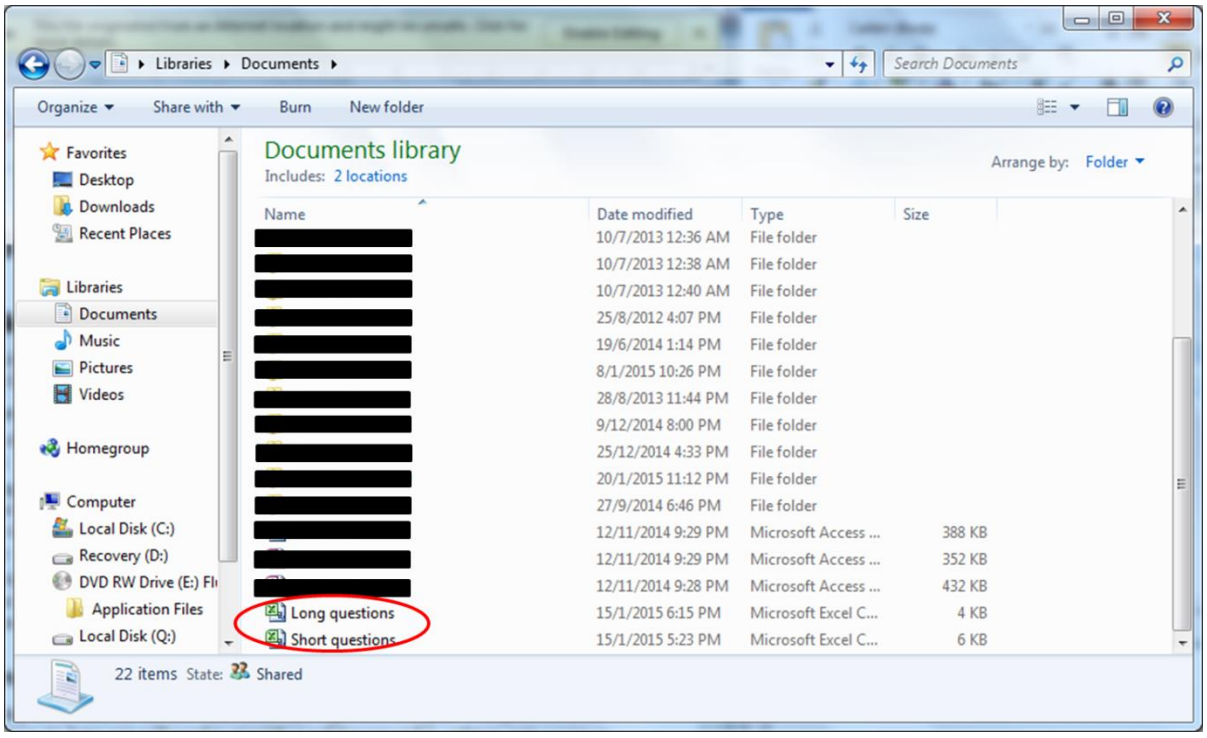


6. Close the application program

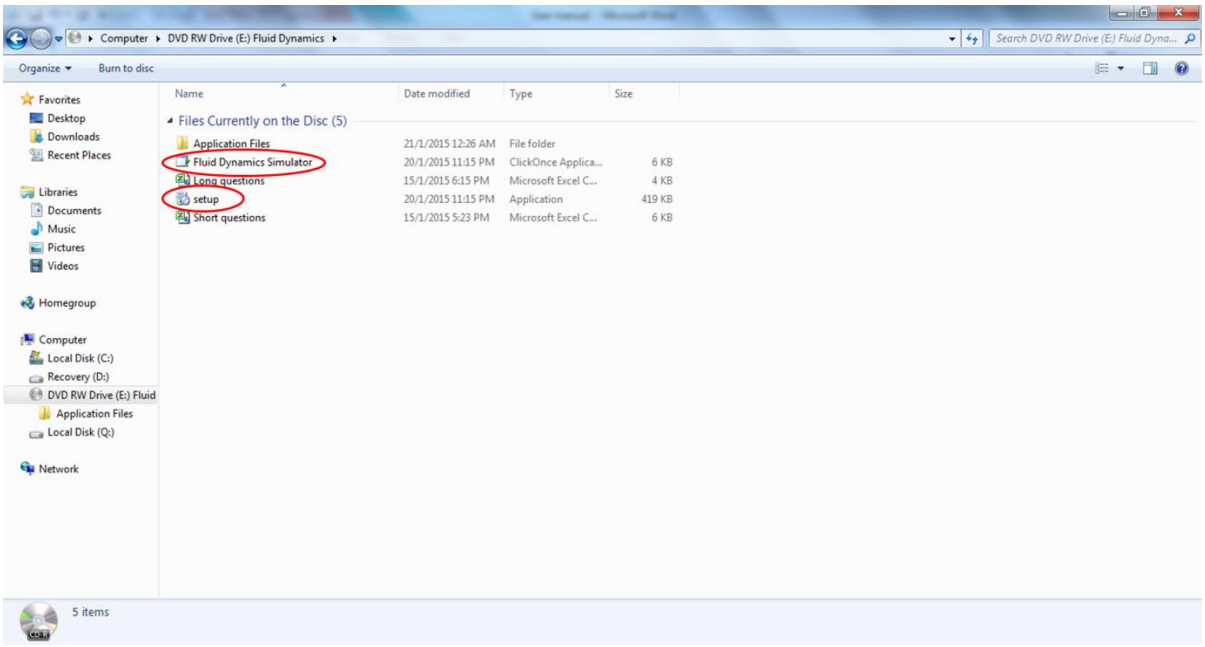
7. Then copy the “Long questions.csv” and “Short questions.csv” files



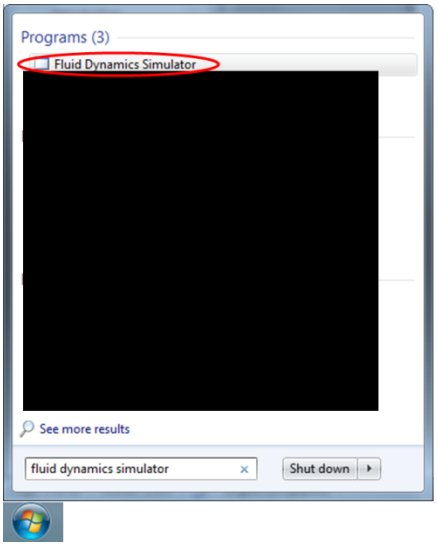
8. After that, paste them in My Documents



9. Now you are able to run the program by clicking on either the setup application or the fluid dynamics simulator clickOnce application deployment manifest.

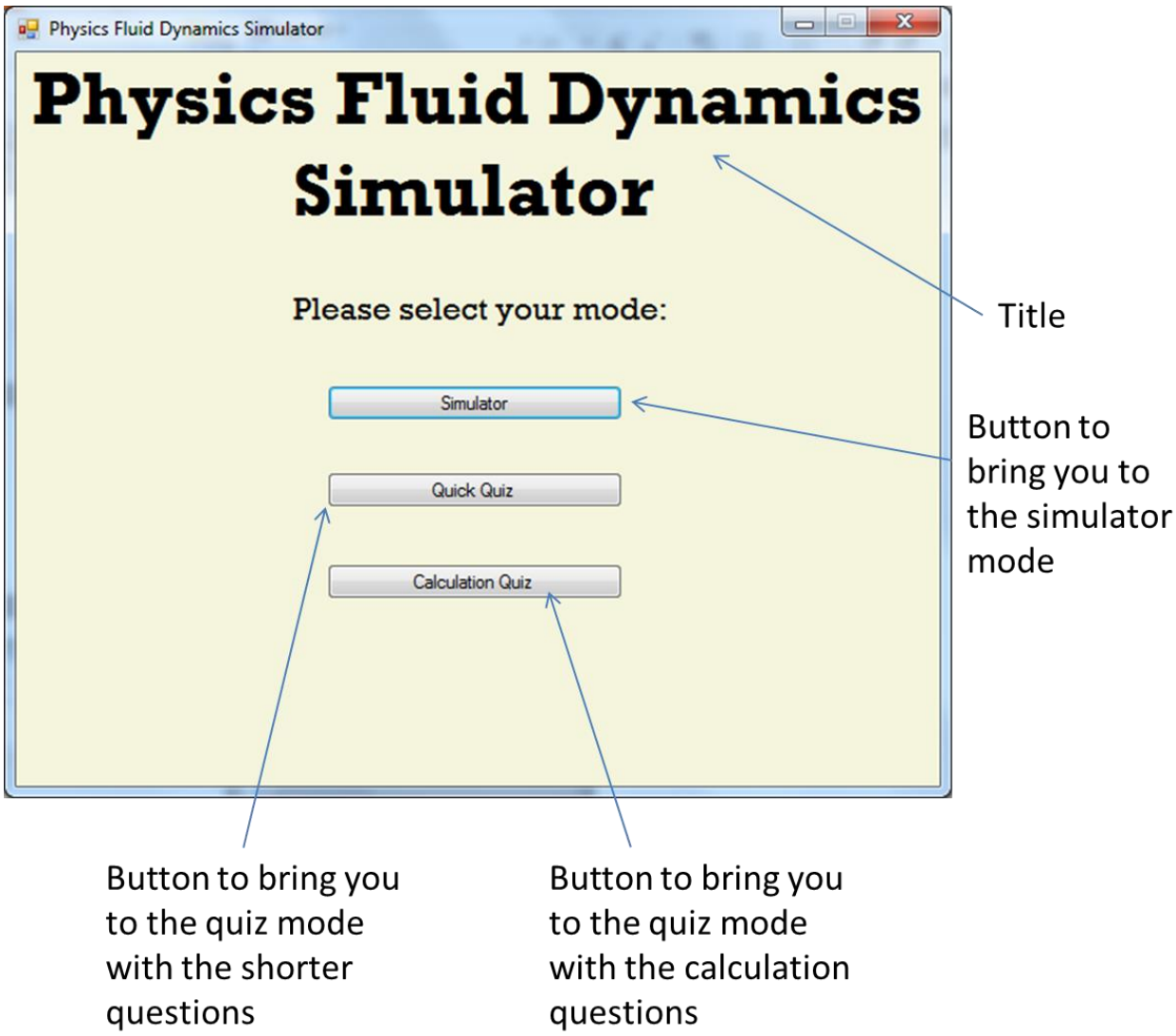


10. To find the program for later use, you can just type in “Fluid dynamics simulator” in the search bar for programs and files and it should appear.



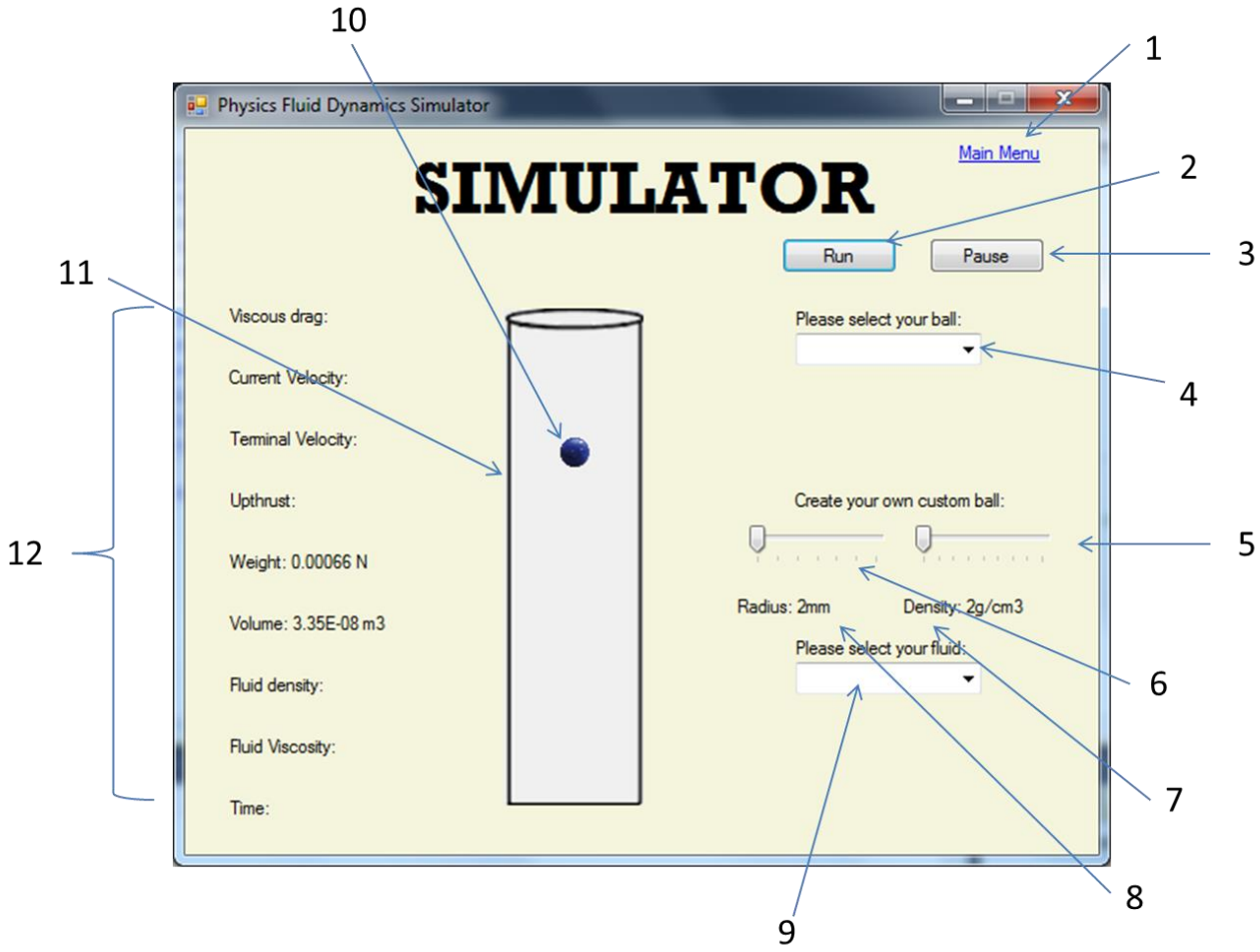
3. Using the Main Menu

Upon launching the application, the first form it should bring you to is the main menu form:



It is very easy to navigate around the page. As you can see, there are only 3 buttons. Each one is clearly labelled and there is a prompt asking you to select one of them.

4. Using simulator mode

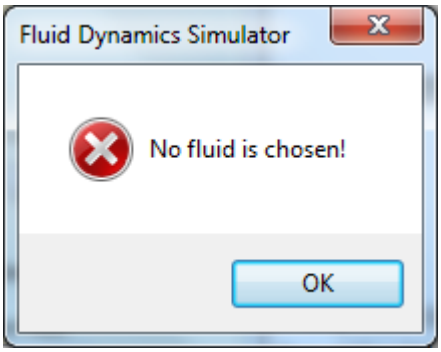


In the simulator mode, you are able to experiment with the movement of the ball by changing the conditions. The simulator interface allows you to change the density and the radius of the ball by setting them to your own liking individually or selecting a pre-defined ball from the drop down menu. It also allows you to select a fluid from a set of pre-defined fluids. You are able to run and pause the simulation, and also go back to the main menu to access the other modes.

I will explain each of the functions labelled on the interface below:

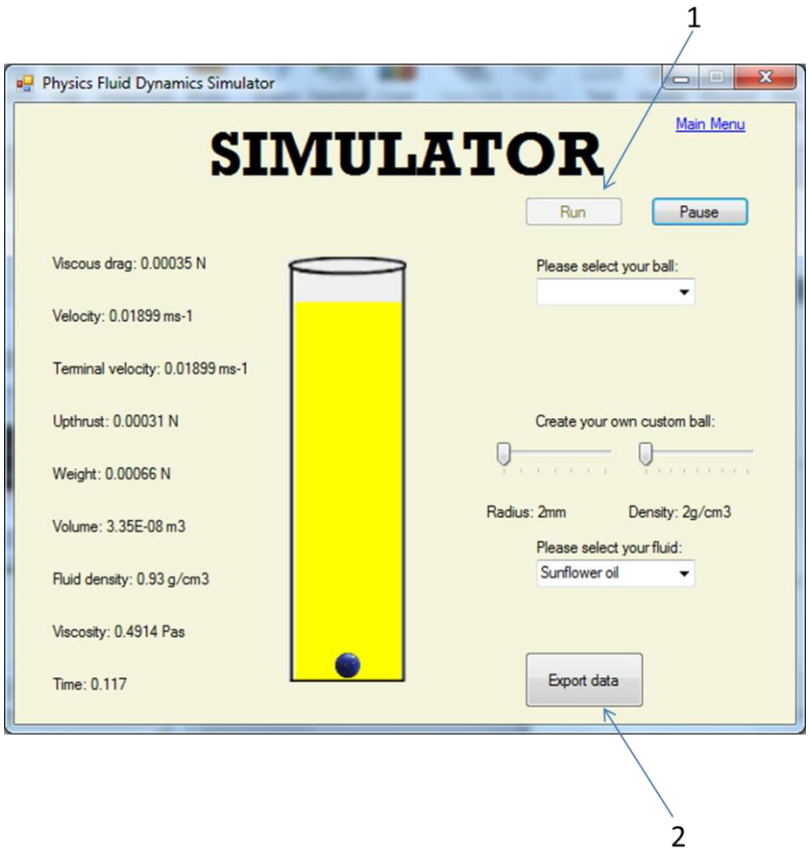
1. **Main menu link –** This link once clicked brings the user back to the main menu, where they can choose a different mode.
2. **Run button –** This button will run the simulation when it is clicked.
3. **Pause button –** This button will temporarily pause the simulation until the run button is clicked again.
4. **Select ball drop down list –** Clicking on this drop down list would produce a list of balls with different radius and density, i.e. 8mm steel ball, 6mm steel ball, 8mm marble ball and 6mm marble ball.
5. **Density slider –** This changes the density of the ball. Its new density can be seen on label 7 and its effects can be seen in the Weight label on 12.

6. **Radius slider –** This changes the radius of the ball. Its new radius can be seen on label 8 and the size of the picture of the ball will change too as the value of the slider is being adjusted.
7. **Density label –** This outputs the value on the density slider (label 5)
8. **Radius label –** This outputs the value on the radius slider (label 6)
9. **Fluid drop down list –** Clicking this drop down lists all the fluids available to them. The user is able to choose from a list of: glycerol, sunflower oil, coconut oil and olive oil. Once a fluid is chosen, the colour of the tube (label 11) will change. You have to choose a fluid first before the simulation can be run. If you don't, then an error message like this would appear.



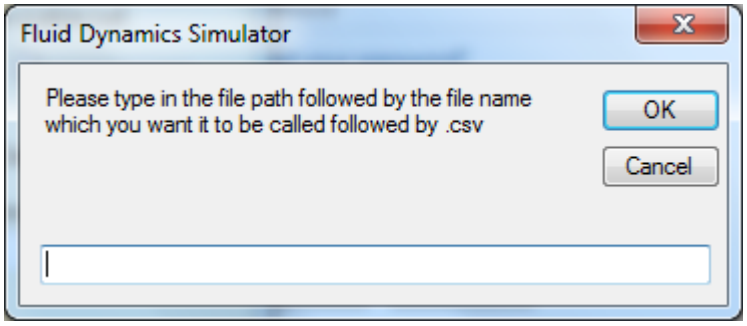
10. **Ball –** This shows a picture of the ball which has radius 2mm and 2gcm^{-3} by default. It is the object that will move once the simulation is run. As you vary the radius slider (label 6) or change the ball from the ball drop down list (label 4) the size of it will change. If you pick a ball of different material from the ball drop down menu (label 4), then the picture of the ball will also change to represent that material. If the density slider (label 5) is then adjusted, the picture of the ball will change to a blue ball which can represent any density.
11. **Tube –** This shows the image of a tube containing the ball. A default tube with no fluid inside is chosen at first. The image of the tube changes as a different fluid is selected from the fluid drop down menu (label 9) because the colours of the fluids are different.
12. **Labels –** These labels show the value each of the variables has. The viscous drag, current velocity, terminal velocity and time labels would output their values once the simulation is run. The viscous drag, current velocity and time values would keep on updating as the simulation is run. The weight and volume values would change as a different ball is selected. The weight value would also change as the value of the radius slider and density slider is changed and the volume value would change as the value of the radius slider is changed. The fluid density and fluid viscosity values would change as a new fluid is chosen.

5. Simulator mode when a run has finished

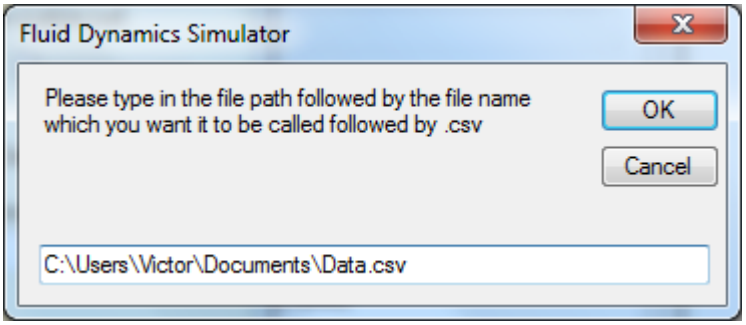


There are some changes on the simulator form once the simulation has ended (i.e. once the ball has reached the bottom of the test tube). Below I will explain the following features that have changed:

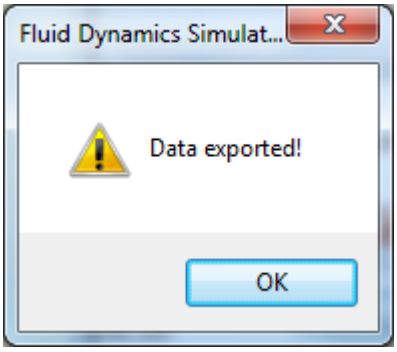
1. **Run button -** The run button is disabled. To enable this button again, you must either:
    - Select a new ball from the select ball drop down list
    - Change the value of the radius slider
    - Change the value of the density slider
    - Change the fluid from the fluid drop down list
2. **Export data button** – Clicking on this button will enable you to export the data from the run you just had. Be sure not to change any of the variables (e.g. the radius of the ball) before clicking on it as this will create null outputs instead. After clicking on this button you will see an input box prompting you where you want the file to be saved and the name of the file.



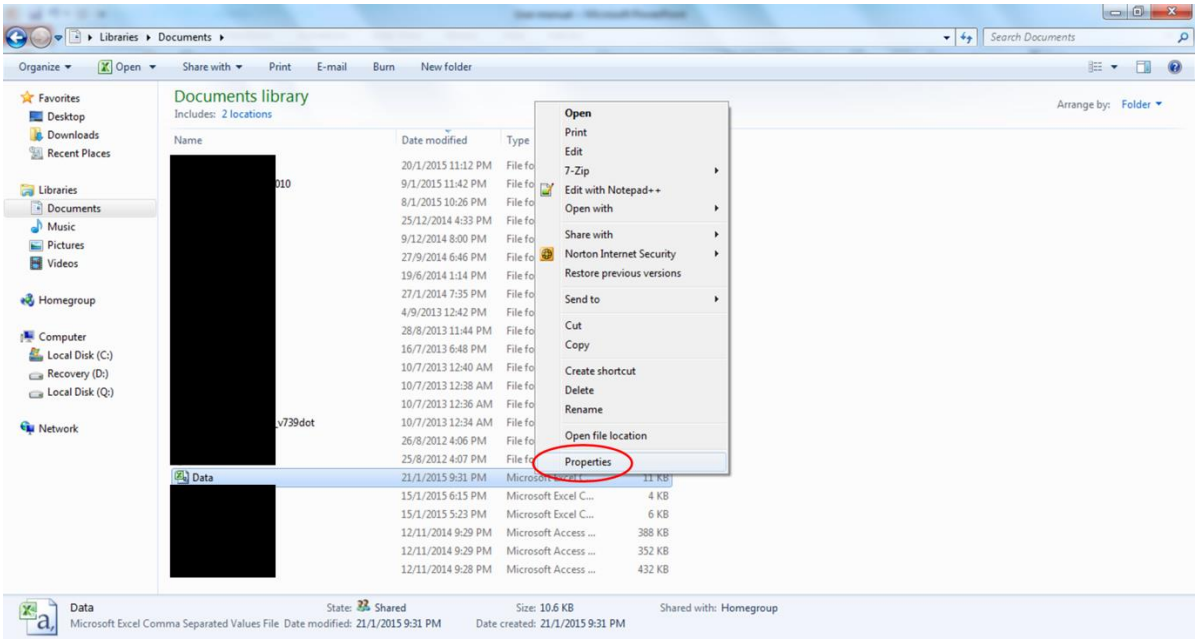
Type the file path and \“file name.csv” into the textbox below e.g.



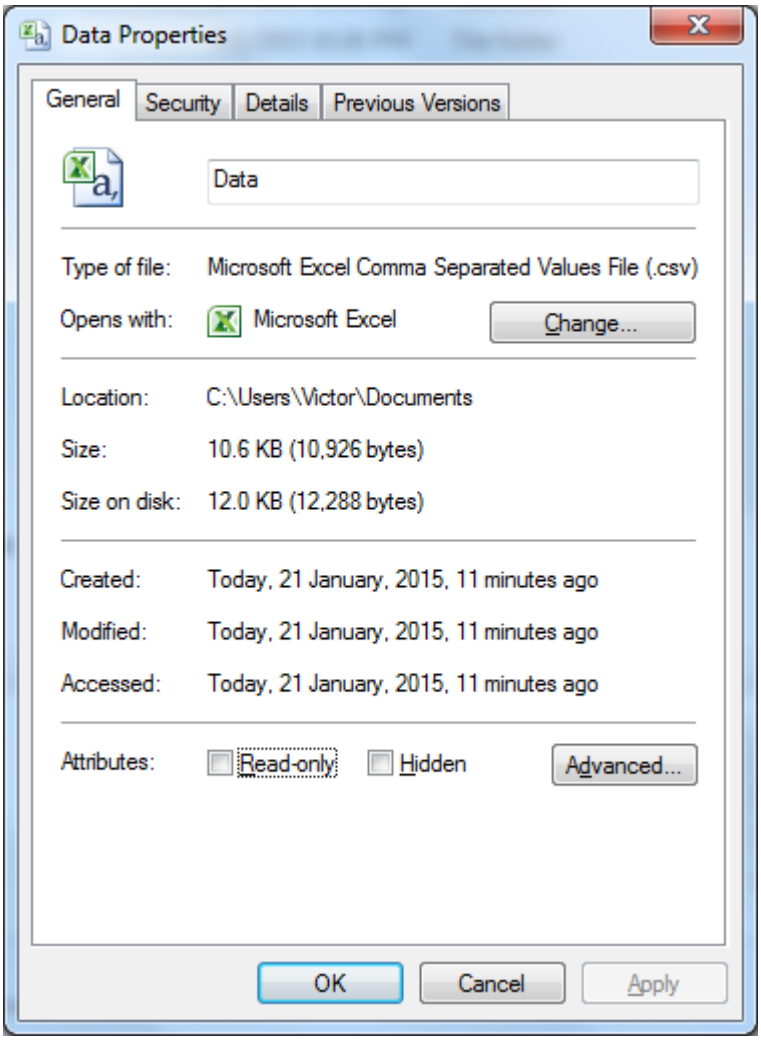
After typing the file path and file name in, click OK. An output message should notify you that the data has been exported successfully.



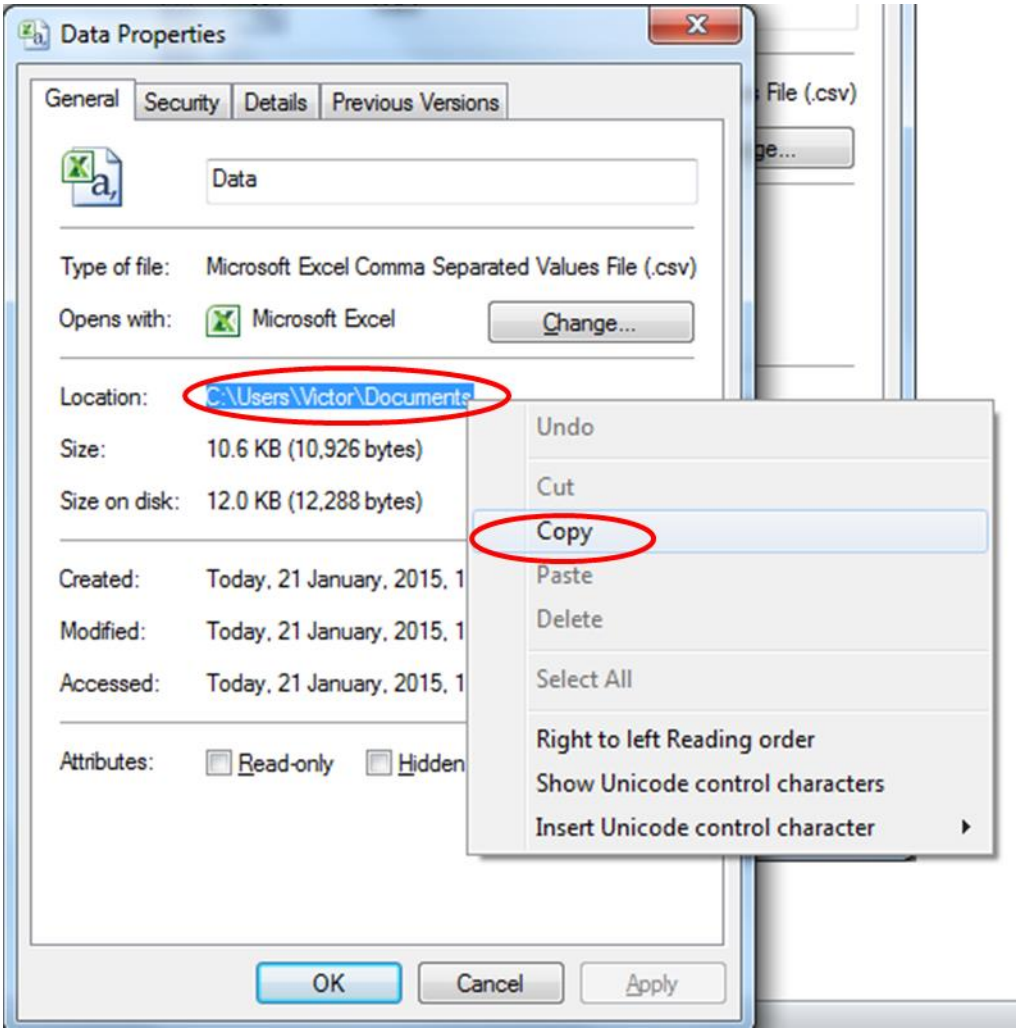
If you want to get the file path quickly then what you can do is right click on a document in that folder and click on properties



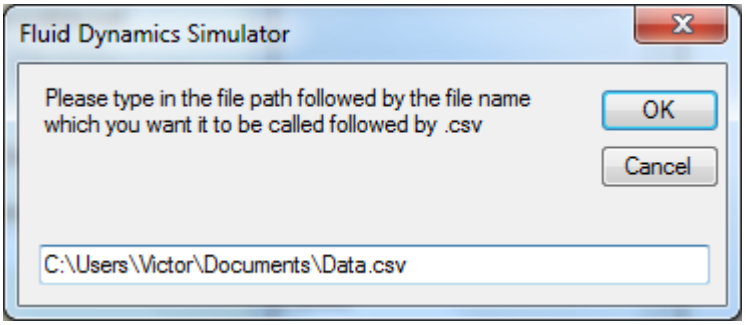
This should bring you to a page which looks something like this:



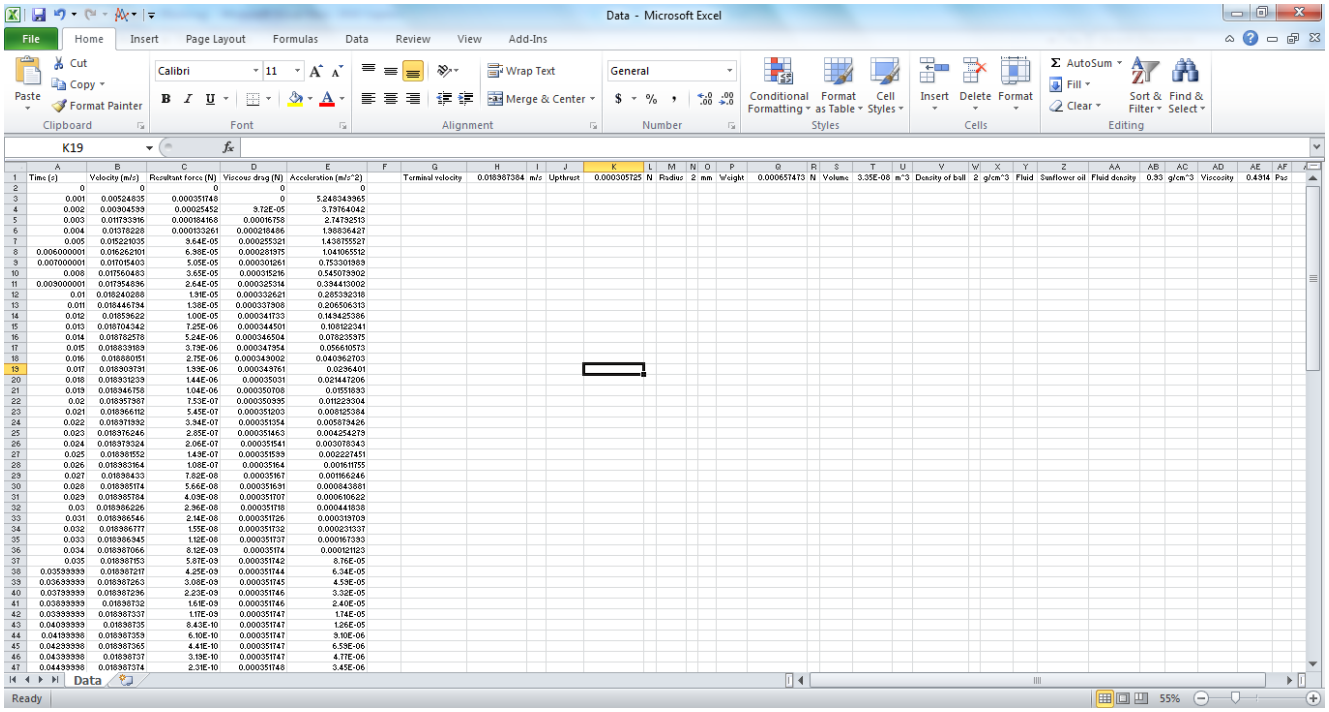
Highlight and copy the file location.



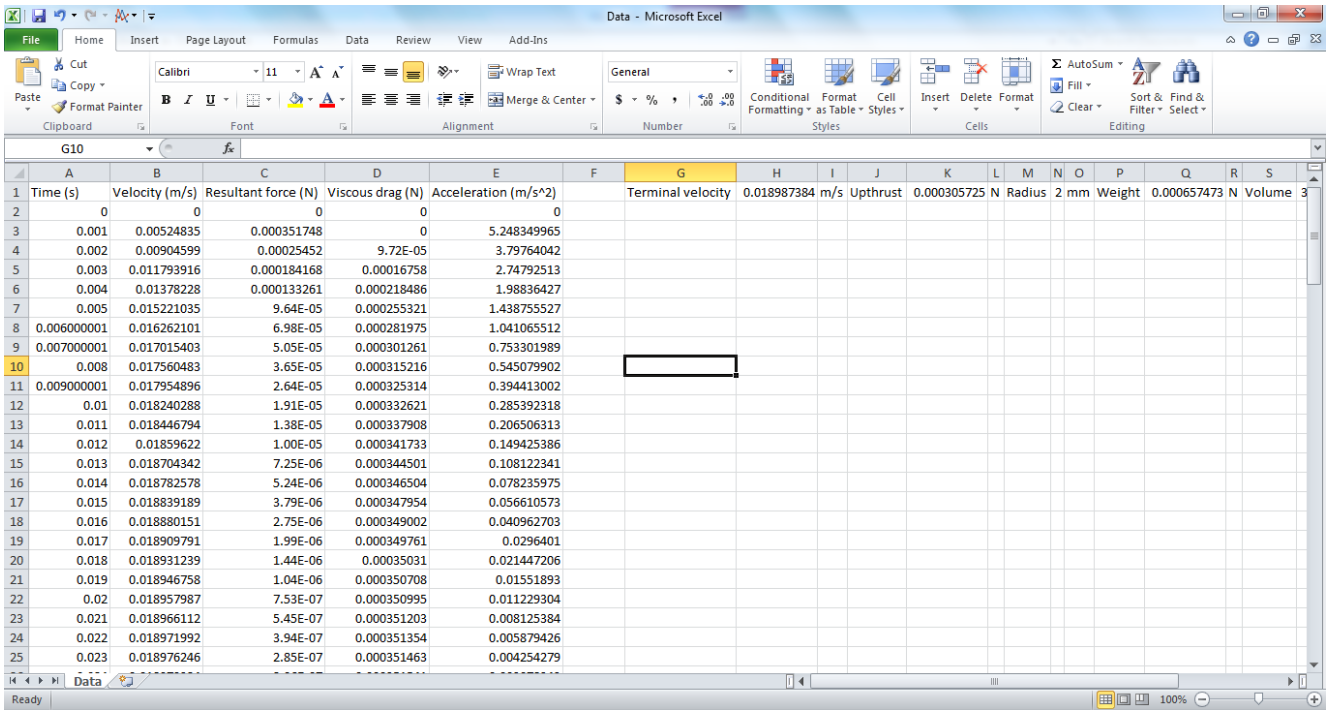
Paste it into the input box and then follow the instructions as before (put \“file name.csv”)



The typical exported data should look something like this:

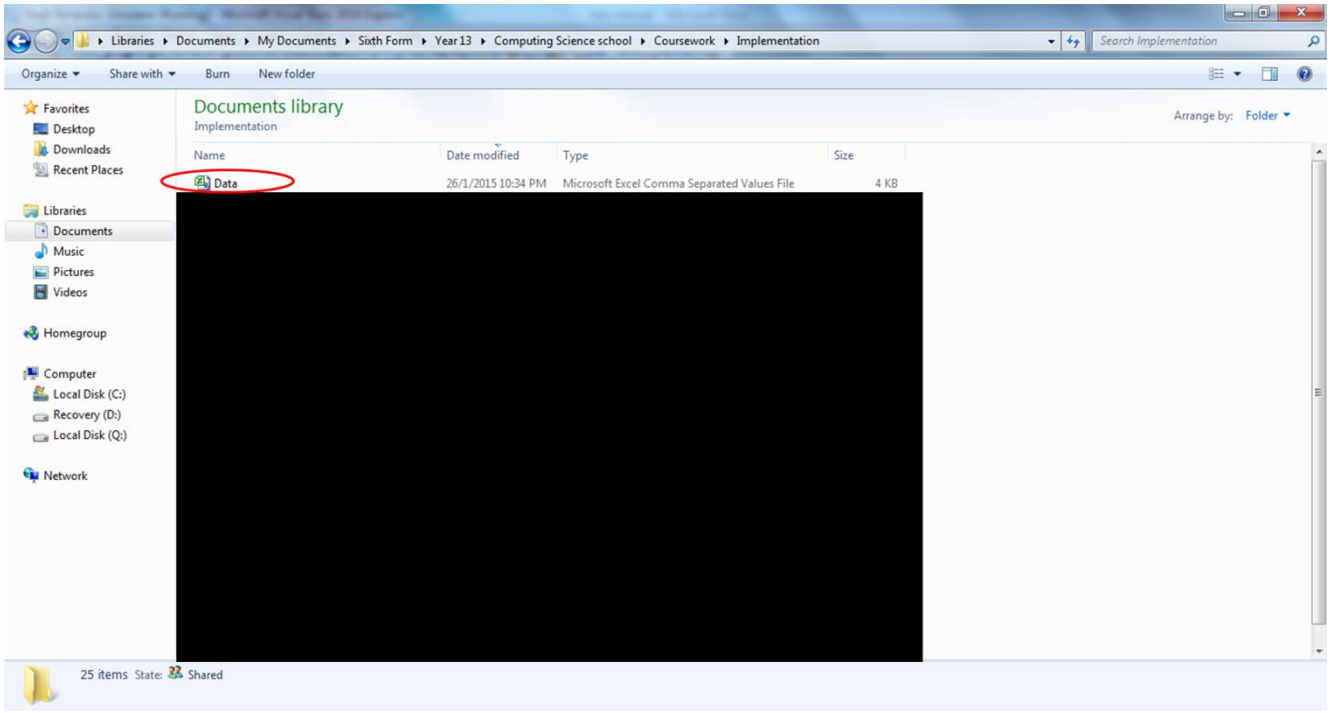


A zoomed in version of what the exported csv file looks like:

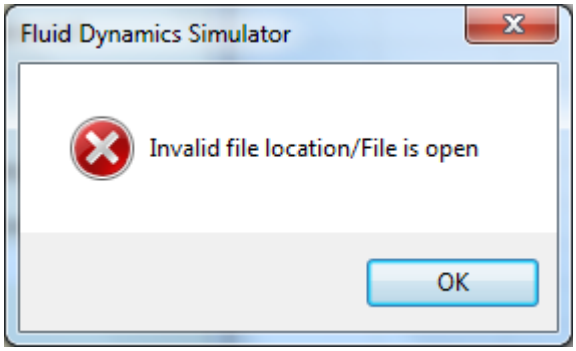


	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Time (s)	Velocity (m/s)	Resultant force (N)	Viscous drag (N)	Acceleration (m/s^2)		Terminal velocity	0.018987384	m/s	Upthrust	0.000305725	N	Radius	2	mm	Weight	0.000657473	N	Volume
2	0	0	0	0	0														
3	0.001	0.00524835	0.000351748	0	5.248349965														
4	0.002	0.00904599	0.00025452	9.72E-05	3.79764042														
5	0.003	0.011793916	0.000184168	0.00016758	2.74792513														
6	0.004	0.01378228	0.000133261	0.000218486	1.98836427														
7	0.005	0.015221035	9.64E-05	0.000255321	1.438755527														
8	0.006000001	0.016262101	6.98E-05	0.000281975	1.041065512														
9	0.007000001	0.017015403	5.05E-05	0.000301261	0.753301989														
10	0.008	0.017560483	3.65E-05	0.000315216	0.545079902														
11	0.009000001	0.017954896	2.64E-05	0.000325314	0.394413002														
12	0.01	0.018240288	1.91E-05	0.000332621	0.285392318														
13	0.011	0.018446794	1.38E-05	0.000337908	0.206506313														
14	0.012	0.01859622	1.00E-05	0.000341733	0.149425386														
15	0.013	0.018704342	7.25E-06	0.000344501	0.108122341														
16	0.014	0.018782578	5.24E-06	0.000346504	0.078235975														
17	0.015	0.018839189	3.79E-06	0.000347954	0.056610573														
18	0.016	0.018880151	2.75E-06	0.000349002	0.040962703														
19	0.017	0.018909791	1.99E-06	0.000349761	0.0296401														
20	0.018	0.018931239	1.44E-06	0.00035031	0.021447206														
21	0.019	0.018946758	1.04E-06	0.000350708	0.01551893														
22	0.02	0.018957987	7.53E-07	0.000350995	0.011229304														
23	0.021	0.018966112	5.45E-07	0.000351203	0.008125384														
24	0.022	0.018971992	3.94E-07	0.000351354	0.005879426														
25	0.023	0.018976246	2.85E-07	0.000351463	0.004254279														

And the file type is of CSV format:



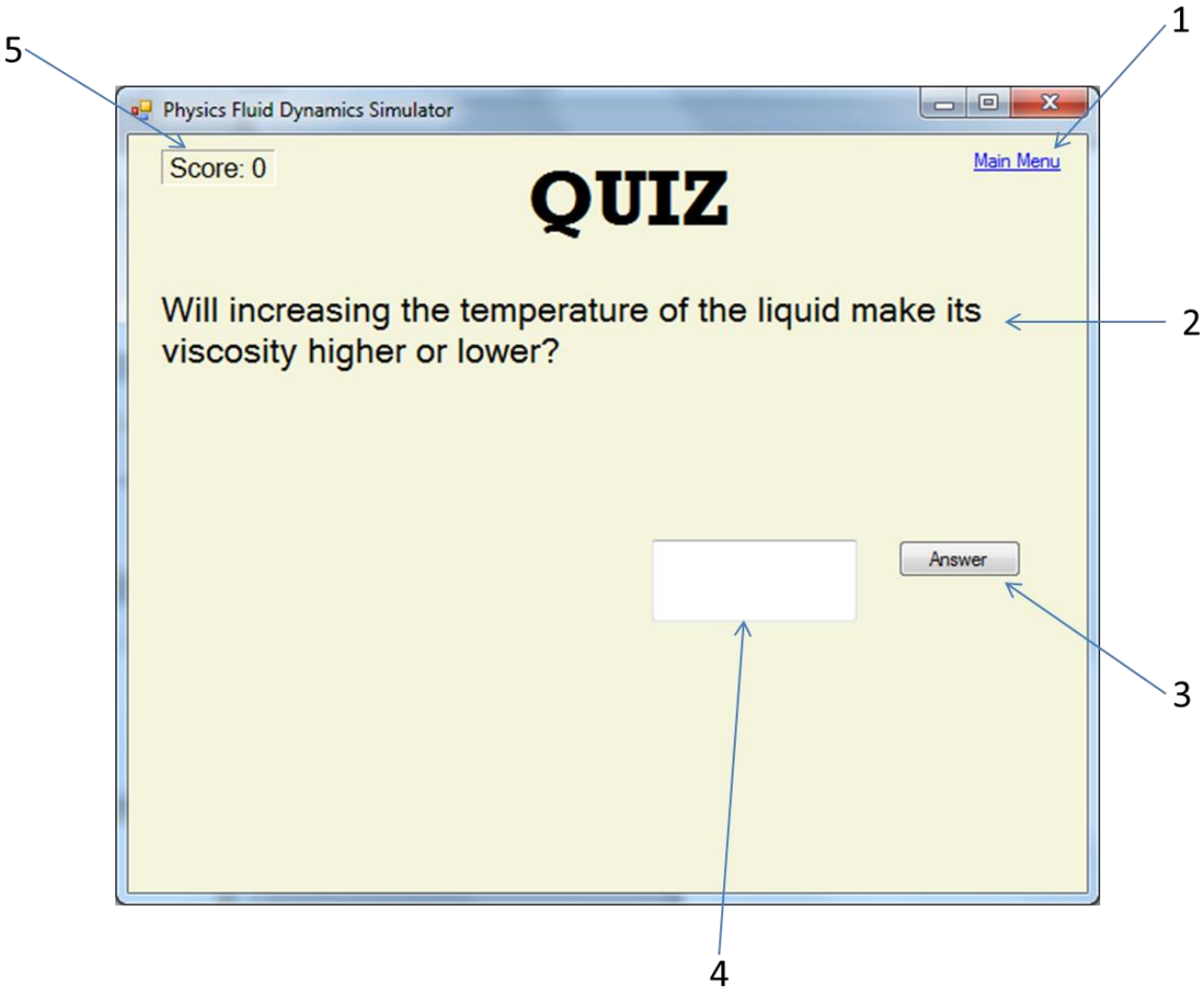
If you want to rewrite to a currently existing file and it is open, you should see an error message when you try to export it to that location.



Close the file and try again.

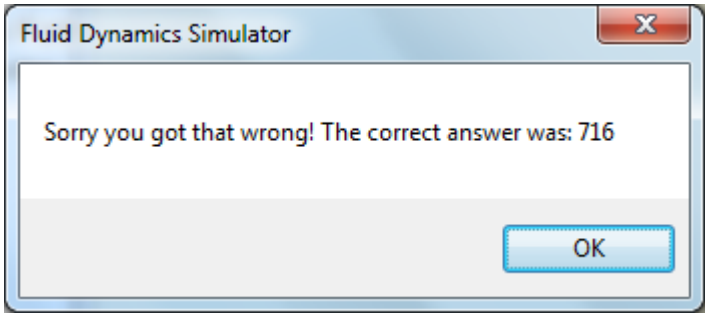
6. Quiz mode

In the quiz mode, you are able to answer questions related to the fluid dynamics topic. The user interface for both quiz modes are the same. The user interface is fairly simple with a textbox for you to type in your answer, an answer button for you to check your answer and a main menu link to go back.

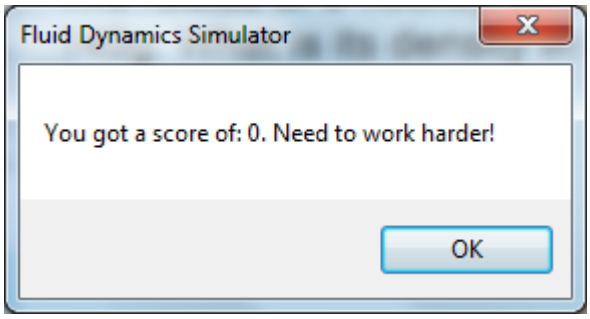


1. **Main menu –** This is a link which takes you back to the main menu where you can choose to go to a different mode.
2. **Question –** This label displays the current question for you to answer.
3. **Answer button –** Once you have finished typing your answer in the answer textbox (label 4) press the answer button to check whether you have got it right or wrong.
4. **Answer textbox –** Type in what you think is the correct answer here.
5. **Score –** This will display your current score depending on how many questions you got right.

The quiz stops when you get an answer wrong or when you have finished answering all the questions available. If you get a question wrong, 2 display messages will be outputted which will look something like this.

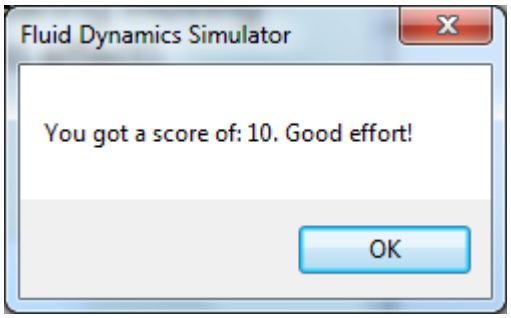


A display showing the correct answer

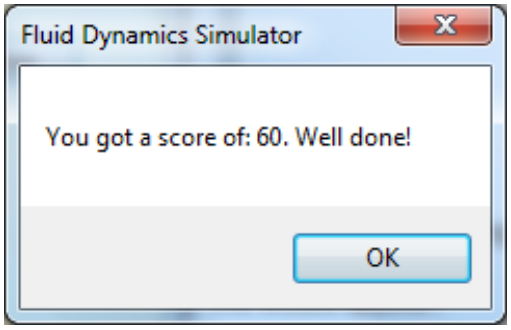


A display showing the final score of the user and a message depending on what score the user got

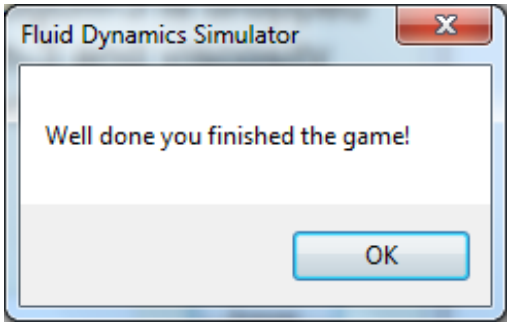
If the user gets a score of over 0, a display like this would appear:



If the user gets a score of over 50, a display like this would appear:



If you finish the game, a display like this would appear:

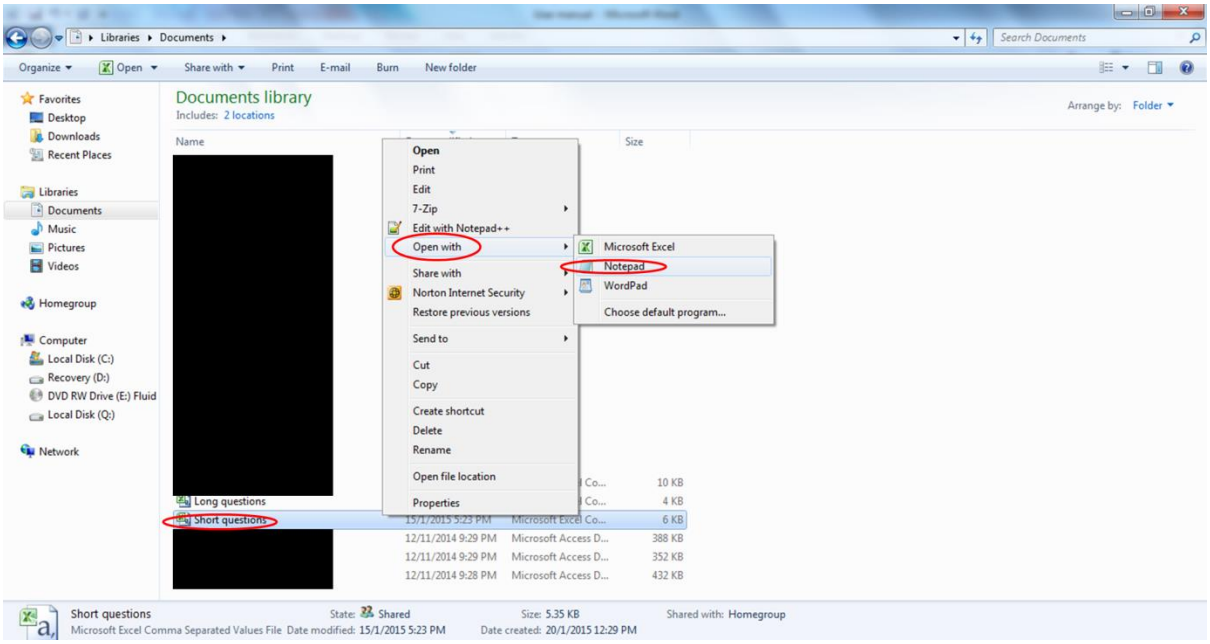


7. Teacher's guidance

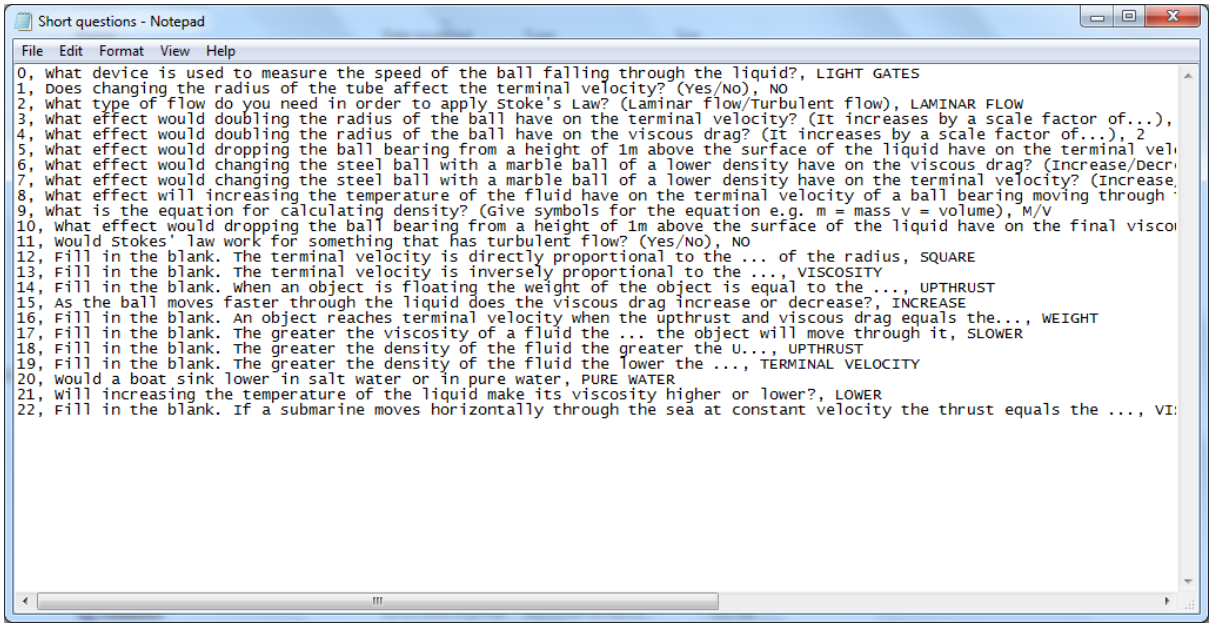
This section is intended for teachers who wish to change the questions or answers for the quiz.

To change the questions or answers for the quiz:

1. Right click on the file you want to change e.g. Short questions.csv then go to "Open with" and click notepad.



2. After that you should see a file like this:

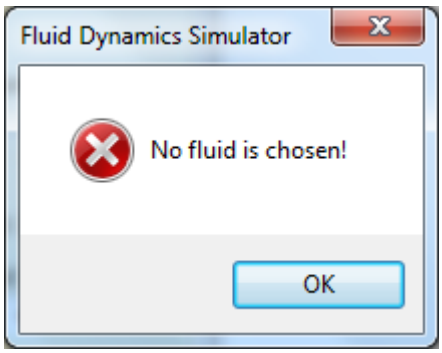


3. To change a question, delete and type in the new question or edit the question. Make sure there is a space after the comma and the question. (N.B. the question cannot contain any commas.)
4. To type in the new answer, put a space between the comma at the end of the question and where you are going to type in your answer
5. To add a new question, simply enter a new line with the question number, followed by a comma, followed by a space, followed by the question, followed by another comma, followed by another space, followed by the answer.

N.B. The question numbers don't actually matter. They are just there to help with the editing of the user; however it is important that there must be something (e.g. the question number) with a comma followed by a space before the question appears.

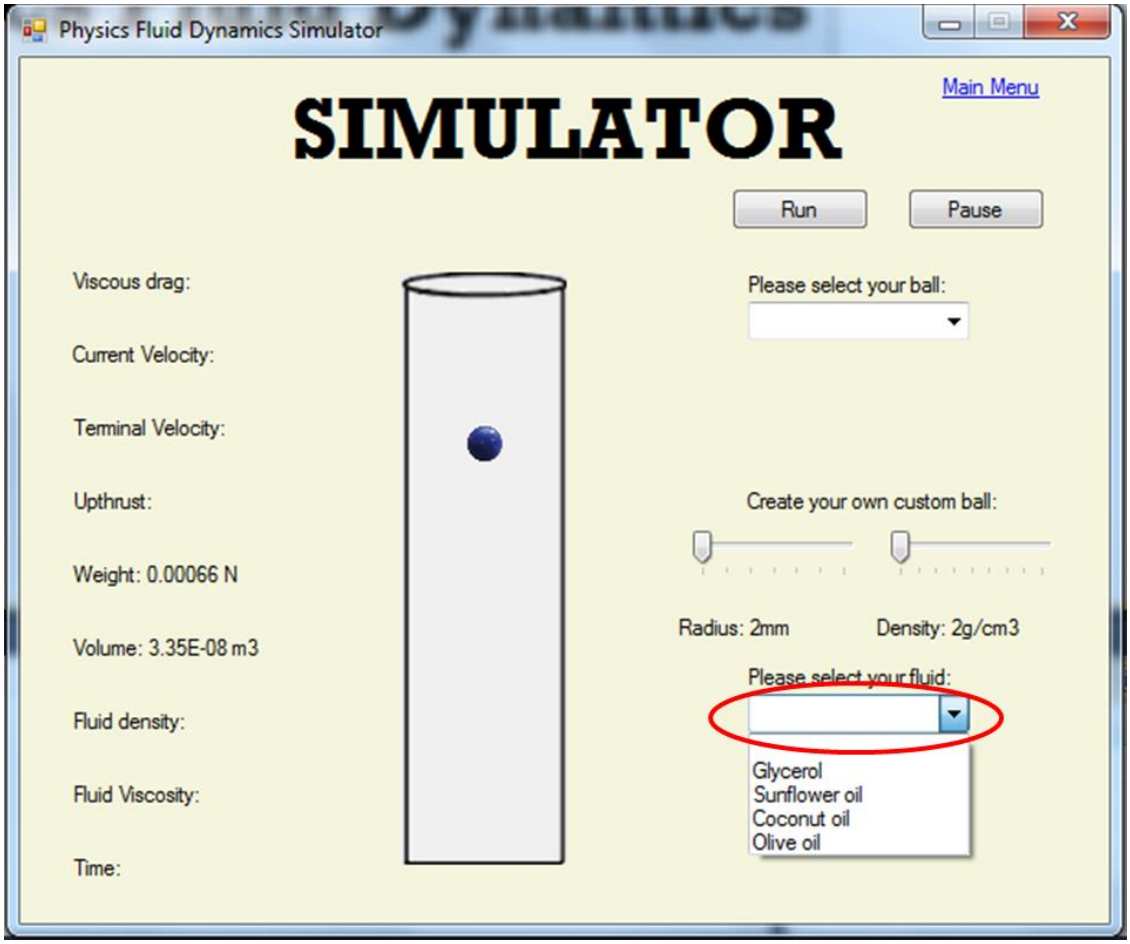
8. Troubleshooting tips

1. If you cannot run the simulation, it is probably because you have not selected a fluid. If you try running the simulation without a fluid chosen, you will get an error message like this:

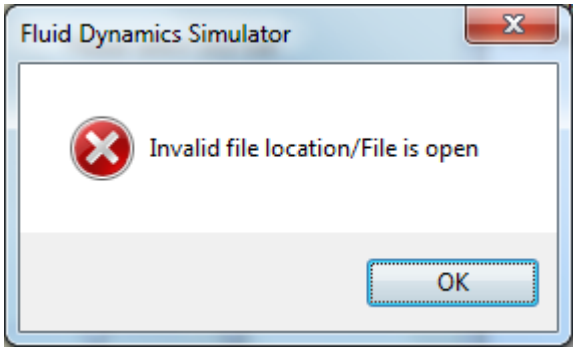


If this happens, simply:

i. Click OK
ii. Choose a fluid from the fluid drop down list



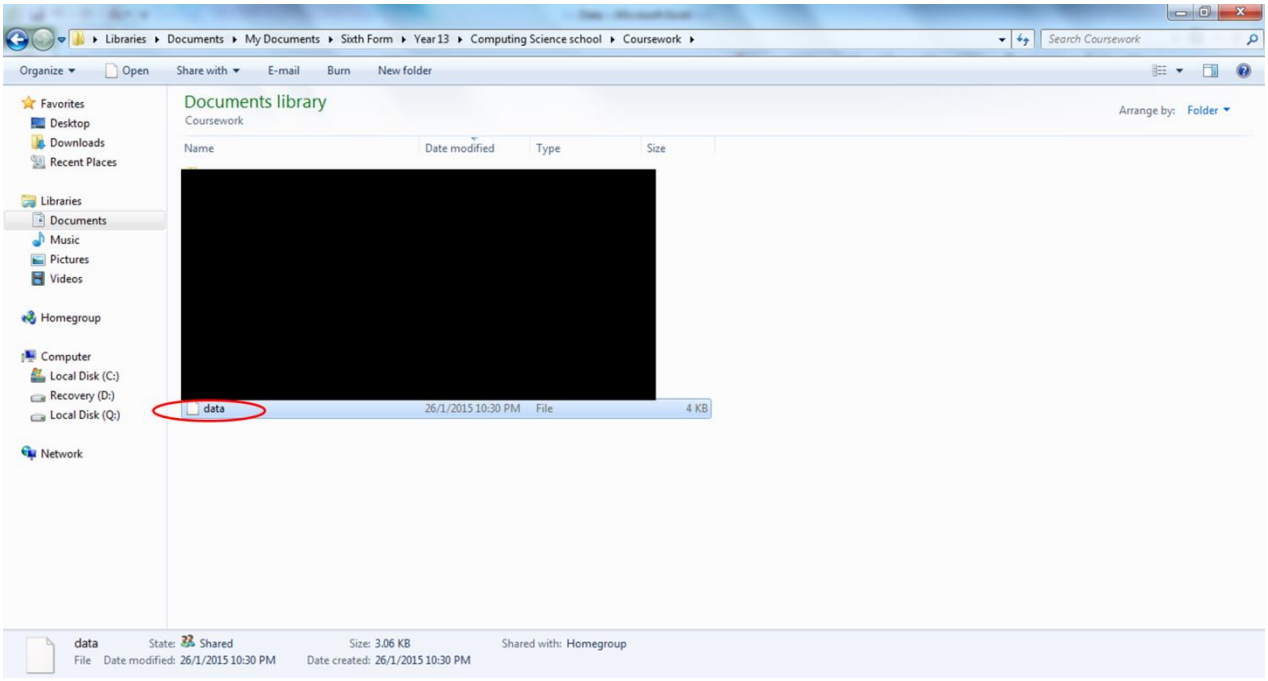
2. If you cannot export the data, then you are most probably overwriting to an existing file which is currently open. An error message will pop up if you try to do this.



Close the file and try again.

3. If you have successfully exported the data but cannot find the file, make sure you have typed in the file location properly.

4. If you have exported the data successfully but the file looks something like this:



Then you have most probably forgotten to put the .csv at the end of your file name. Try exporting the data file again keeping in mind of the .csv at the end.

5. If you cannot play the quiz, then you probably have forgotten to copy and paste the question files into your documents. Follow steps 7 and 8 of the installation process and try again. An example of how the quiz mode would look like if the files aren't in the right place would look like this: