

VIRGINIA VARIABLES

Overview

Pupil explore how variables can be used in different ways before creating their own programs that use variables, sequences and loops.

To do before the session

- 1, Print pupil worksheets in colour double sided and staple into a booklet one for each student
- 2, Print 10 marksheets, sort into piles places where pupils can access them independently
- 3, Download the code needed and place in a templates folder on your school network or add to a Scratch Studio or link on your learning platform.
- 4, Download the variables slides that go with the concept introduction
- 5, Study the notes that go with the variables slides (if you have the Year 6 book)

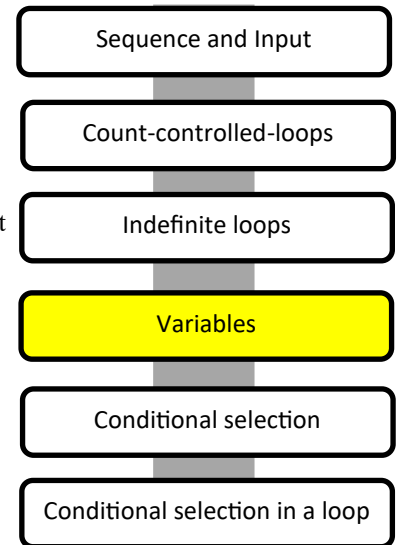
To do at the start of the session

If you have not introduced **variables** with this class before do this first using the variable slides

To do after the concept has been introduced

It can also be helpful to show pupils how to create a variable if they haven't already done so and give them 20 mins to experiment with creating and using a variable without any directed theme. However useful this time is don't be tempted to believe that pupils have good understanding even if they can use variables in simple ways. They will need careful code comprehension in PRIMM to broaden their understanding.

How this module fits into a programming progression



Resource Name	Core Optional SEN	Teacher	Pupil Grouping	How Assessed	SCRATCH ACCESS
CONCEPT Variables that change	CORE	Leads Session	Solo whole class activity	Formative	NO
PREDICT	CORE	Support Poor Readers	Paired	Pupil Marked Marksheet Provided	NO
INVESTIGATE	CORE	Support Poor Readers	Paired	Pupil Marked Marksheet Provided	YES Variable fun
CHANGE	CORE	Support Poor Readers	Paired	Pupil Marked Marksheet Provided	YES Variable fun
CREATE	CORE	Assesses pupil work and checks pupil self assessment	Solo	Pupil Assessed & Teacher Assessed	YES Virginia Variables

Vocabulary
variable, assign, set, value, name

Core activities general instructions

1, Group pupils in roughly same ability pairs. For **investigate** and **change** worksheets pupils will work in pairs, for **create** they will work separately.

2, Give out the pupil booklets and explain that pupils need to follow the instructions on the sheets to explore how **variables** work.

3, Explain that each pupil will record separately whilst working alongside their partner and keeping to the same pace as their partner

4, Demonstrate where they can find the template code and explain that pupils will share one device for investigate and change but have a device each when they get to make

5, Explain that during each question only one person should touch the shared device and they should swap who that person is when there is a new questions.

6, Encourage them to discuss their answers with their partner. If they disagree with their partner they can record a different answer in their own booklet.

7, Show pupils where it says they should mark their work on the sheet and where the answer sheets are in the classroom.

8, Remind pupils to return marksheets after marking because there are not enough for every pair to have their own.

Key Programming Knowledge

Variables are used to store information to be referred to and changed in a computer programme or algorithm

Variables

- Have a name and a value
- read the name but act on the value
- Values can be changed during the algorithm or programme
- When writing the value of a variable we call it assigning

Variable Naming

- Always name a variable after the data that it stores or the task that it does
- Avoid naming variables with spaces teamScore (camelCase) user_name (underscore) are good methods to use

NOTE Scratch allows multiple word variables but few other programming languages do so it is better to get into good habits

Resources

Virginia Variables <https://scratch.mit.edu/projects/901180358/>

Virginia Computing Science Standards

Grade 4.1 The student will construct sets of step-by-step instructions (algorithms) both independently and collaboratively a. using sequencing; b. using loops; c. using variables to store and process data; and d. performing number calculations on variables (e.g., addition, subtraction, multiplication and division).

Grade 4.2 The student will construct programs to accomplish a task as a means of creative expression using a block- or text-based programming language, both independently and collaboratively a. using sequencing; b. using loops; c. using variables; and d. performing number calculations (e.g., addition, subtraction, multiplication and division) on variables.

Grade 4.3 The student will analyze, correct, and improve (debug) an algorithm that includes sequencing, events, loops and variables.

Grade 4.4 The student will create a plan as part of the iterative design process, independently and/or collaboratively using a variety of strategies (e.g., pair programming, storyboard, flowchart, pseudo-code, story map).

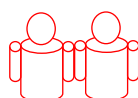
Grade 4.6 The student will break down (decompose) a larger problem into smaller sub-problems, both independently and collaboratively.



On the sheet if it says no Scratch they must work only on the sheet,



If it says Scratch with a green tick they can use Scratch



If it says work with a partner they must work at the same speed as their partner.



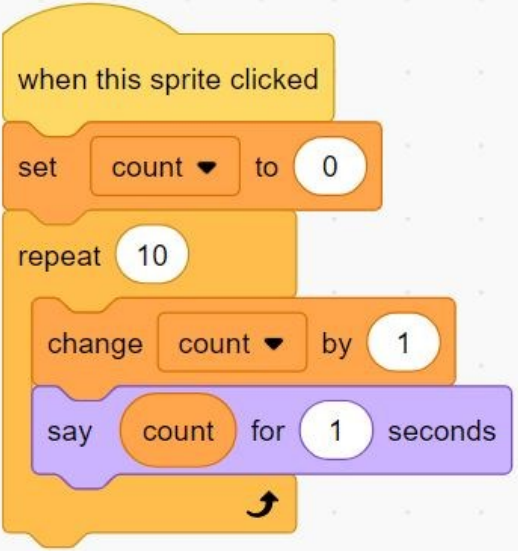
If it says work on their own they must do this using a separate device pairs can still discuss their work

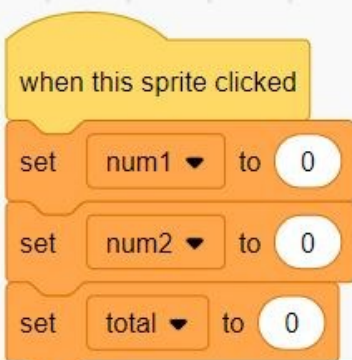

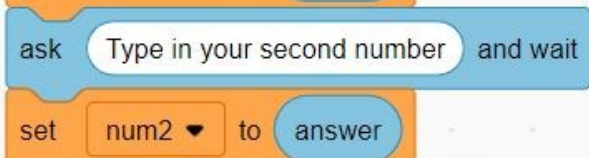
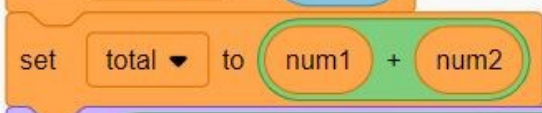
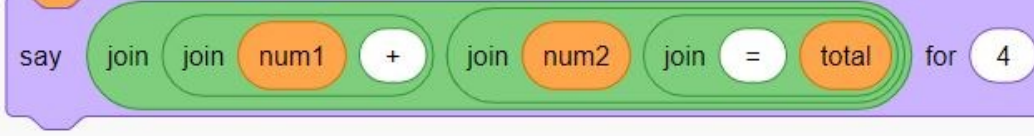
VIRGINIA VARIABLES

PREDICT

Read the code and the code labels carefully with your partner

Write in the box what you think the code does

Assigns 0 to count variable (Initialisation)		I predict this code will When it is run
Adds 1 to count variable		
Says what is in the count variable		

Assigns 0 to all three variables (Initialisation)		I predict this code will When it is run
Asks the users to type in a number and then assigns that number to num1		
Asks the users to type in a number and then assigns that number to num2		
Adds num1 and num2 and puts the answer into total		
Tells the user what is inside all three variable		

Now mark your work using the predict marksheet

VIRGINIA VARIABLES

INVESTIGATE



Work with a partner
of same ability

Work with a partner. Open Scratch and load
Virginia Variables

Run the program as many times as you want. Work with a partner to answer these questions.

Look inside the Placeholder sprite

1, Which two lines of code collect the users name and assign it to the variable **user_name**?

A

B

2, How many times is the **user_name** variable block used in the program?

Look inside Accumulator sprite

3, Which block of code adds one to the score variable?

4, How many times is **switch costume to andie-c** run?

5, Which block of code tells everyone what the score is?

Look inside Adding sprite

6, Name all three variables

A,

B,

C,

7, Which line of code blocks adds num1 to num2?

Look inside Counting

8, What line of code says what is inside the count variable?

9, Which block initialises the count variable?

HINT Sets it back to what it was at the start

10, How many times is say count by 1 run?

Now mark your work using the investigate marksheet



VIRGINIA VARIABLES

CHANGE

Work with a partner. Open Scratch and load
Virginia variables



Make changes to the code to answer these questions

Make changes to Accumulator

1, Change the code to give 2 points for every time a basket is scored.

What did you change?

2, Change the code to start the score at 9 points

What did you change

Make changes to Adding

3, Change the code to multiply both numbers

Circle the block you changed it to?



Make changes to Counting *HINT Duplicate code and then change*

4, Change the code to count slower

What did you change?

5, Change the code to count in 2s.

What did you change?

6, Change the code to count in halves. *HINT decimal halves*

What did you change

7, Change the code to count backwards from 10.

What two things did you change

A,

B,

Now mark your work using the change marksheet

VIRGINIA VARIABLES

CREATE



Work within Virginia Variable to complete tasks 1-4. You can change code or duplicate a sprite and then change the code

Task 1 Placeholder

Add some more say blocks that use the `user_name` variable to the end of the programme to tell the user something interesting

Task 2 Adding

Work within **Adding** to make the program add three numbers together HINT



Task 3 Counting

Plan and make a 30 second count down timer that does something interesting when it gets to 0

Task 4 Adding

Plan and make a programme to subtract one number from another number

Task 5 Create your own programme that uses variables

Objects (sprites and back-grounds) Overview	Algorithm Plan
---	----------------

Teacher and Pupil Assessment

Circle one column on each row to show what you think you have achieved



	Not used a variable in my program	copied a variables idea exactly	Adapted a variable idea	Used variables in a way not shown in the example programs
Variables	0 Marks	1 mark	2 marks	3 marks
			Not used previous programming concepts for real purpose	Used previous programming concepts for real purpose
Used previous programming concept such as loops correctly			0 Marks	1 mark

VIRGINIA VARIABLES

PREDICT

Answer sheet

Assigns 0 to count variable
(Initialisation)

Adds 1 to count variable

Says what is in the count
variable

```
when this sprite clicked
  set count to 0
  repeat 10
    change count by 1
    say count for 1 seconds
```

I predict this code will
Count or count up or count to
10 or 1, 2, 3, 4, 5, 6, 7, 8, 9,
When it is run

Assigns 0 to all three
variables
(Initialisation)

Asks the users to type in a
number and then assigns
that number to num1

Asks the users to type in a
number and then assigns
that number to num2

Adds num1 and num2 and
puts the answer into total

Tells the user what is inside
all three variable

```
when this sprite clicked
  set num1 to 0
  set num2 to 0
  set total to 0
  ask Type in your first number and wait
  set num1 to answer
  ask Type in your second number and wait
  set num2 to answer
  set total to num1 + num2
  say join join num1 + join num2 join = total for 4
```

I predict this code will
Add two numbers together or
do a sum or addition
When it is run

VIRGINIA VARIABLES INVESTIGATE MARKSHEET

Run the program as many times as you want. Work with a partner to answer these questions.

Look inside the Placeholder sprite

1, Which two lines of code collect the users name and assign it to the variable **user_name**?

A Ask what's your name and wait (1 mark)

B set user_name to answer (1 mark)

2, How many times is the **user_name** variable block used in the program? **3 (1 mark)**

Look inside Accumulator sprite

3, Which block of code adds one to the score variable? **Change score by 1 (1 mark)**

4, How many times is **switch costume to andie-c** run? **3 times (1 mark)**

5, Which block of code tells everyone what the score is?

Say join your score so far is user_name for 2 seconds (1 mark)

Look inside Adding sprite

6, Name all three variables

A, num1 B, num2 C, total (3 marks one for each one)

7, Which line of code blocks adds num1 to num2?

Set total to num1 + num2 (1 mark)

Look inside Counting

8, What line of code says what is inside the count variable?

Say count for 1 seconds (1 mark)

9, Which block initialises the count variable?

HINT Sets it back to what it was at the start

Set count to 0 (1 mark)

10, How many times is say count by 1 run? **10 times (1 mark)**

If you get lots of questions wrong and you can't see why they are wrong talk to your teacher

VIRGINIA VARIABLES CHANGE MARKSHEET

Make changes to the code to answer these questions

Make changes to Accumulator

1, Change the code to give 2 points for every time a basket is scored.

What did you change? **change score by 2 (1 mark)**

2, Change the code to start the score at 9 points

What did you change? **set score to 9 (1 mark)**

Make changes to Adding

3, Change the code to multiply both numbers

What did you change it to?



Make changes to Counting *HINT Duplicate and then change*

4, Change the code to count slower

What did you change? **Change say count for 1 second to a higher seconds number**

Or add a wait block into the loop (1 mark)

5, Change the code to count in 2s.

What did you change? **Change change-count-by-1 to change-count-by-2 (1 mark)**

6, Change the code to count in halves. *HINT decimal halves*

What did you change? **Change change-count by 1 to change count by 0.5 (1 mark)**

7, Change the code to count backwards from 10.

What two things did you change?

A, **change set count to 0 to set count to 10 (1 mark)**

B, **change change-count-by 1 to change-count-by -1 (1 marks)**

If you get lots of questions wrong and you can't see why they are wrong talk to your teacher

VIRGINIA VARIABLES

FLOW OF CONTROL

Adding

```
when this sprite clicked
  set num1 to 0
  set num2 to 0
  set total to 0
  ask "Type in your first number" and wait
  set num1 to answer
  ask "Type in your second number" and wait
  set num2 to answer
  set total to num1 + num2
  say join join num1 join num2 join = total for 4 seconds
```

Counting

```
when this sprite clicked
  set rotation style left/right
  point in direction -90
  set count to 1
  say "Counting to 10" for 2 seconds
  repeat 10
    change count by 1
    say count for 1 seconds
```

VIRGINIA VARIABLES

EXAMPLE PLANNING

COUNTING PLANNING

OVERVIEW

Add 1 to count variable in a loop

Say what is assigned to count every time it goes through a loop

ALGORITHM

Set count variable back to 0 (initialisation)

Loop 10 times

Add 1 to count variable

Say what is inside count variable

ADDING PLANNING

OVERVIEW

1. Get user to type in two numbers

2. Assign those numbers to variables

3. Add variables and assign to a new variable

4. Tell user what is inside the all variables

ALGORITHM

Set all the variables back to 0 (initialisation)

Ask user to type in a number and assign to num1 variable

Ask user to type in a number and assign to num2 variable

Add num1 and num2 and assign to total variable

Say num1 + num2 = total