

Q1)-a)-i) the colour turns to black.

a)-ii) The variable stores memory values for use by programs, and the name should indicate its purpose. It is also recommended that you name the variables for their purpose, but short.

b)-i) Because it was declared a parameter of a function. In other words, *shift* is given a constant value, which remains in block {} until the block is finished.

b)-ii) If/else clearly distinguishes between true and false, the else condition statement is finally executed if it is not true in the if condition. However, else if serves to help determine more true/false conditions. In this code, the execution value is not fixed when you enter the keycode, but must continue to move according to the keycode.

c)

d)

e)

Q2)-a)-i) *this*. is used to eliminate confusion between parameters with the same name as the class attribute.

a)-ii) Use the variable *new*. *new* is responsible for creating class-type objects. This means that the new operator allocates space to store the data in memory, returns the reference value of that space to the object, followed by calling the constructor.

a)-iii) The string array called *args[]* is where the options given when the Java program runs are automatically contained. Arguments is the value that you substitute for the parameter. That is, the variable that is actually assigned to memory. The first place in Java to load code is *main()*. When reading the *main()* method code, if you give a value after the class name to the *String[]* array named *args*, the string is truncated based on the space, placed in the array, and moved over to the parameter value of the *main()* method.

b)-i) Using *void* to define the fight method, then *if/else* and *else if* statement are used to position the offense and defense and distinguish which of Monster 1 and Monster 2 will win or lose.

b)-ii)

c)

d)

Q3)-a) The constructor is called when an object in the class is created. Can be used to set initial values for object attributes. Subclasses listed from line 7 through 11 are all about setting the Farmer's properties. *super* is used to refer to the superclass of the subclass where the subclass is *n,ag* and the superclass is Farmer. *double* is the type of data for floating points that are larger than the number that can be stored at a float. *int* is a number and *String* is a character. In particular, create a new object called *t1* on line 11.

b)-i) Value within appropriate range of *int* was given, and *rest* was designated as "void" in the form of a method signature because it did not return a value as a method.

b)-ii) First, create the class using the name of the method. It then generates new object-specific fields for each parameter, local variable, and field used by the method. Create a constructor that treats the parameters of the original method and the fields used by the method as parameters. Create a new class of objects and replace the original method with the method of the newly created object.

c)-i)

c)-ii) This example has 'polymorphism' object orientation feature. Polymorphism is the ability of objects of different classes to behave in their own ways when they receive the same message. When polymorphism works in conjunction with inheritance, it is very powerful. In particular, the generalization relationship not only simplifies the code but also enables flexible response to changes.

d)-i) AnimalCare class needs to be expanded, and a new class created by an extension of the class takes over both the variables and functions of the original class. This is called 'extends'. However, compilation errors can occur during this process. And if this happens, the person class naturally becomes a subclass. However, I think it's better to expand by adding functionality, rather than create a new class.

d)-ii)

e) In Java, *static* is the keyword for setting class (static) members. All members with *static* are available without the creation of separate objects (instances). A static variable of one class is stored in the Class area, so any instance has the same value. *final* has a function that prevents you from changing the value once set. If you try to change it, an exception occurs. In this case, it was used to define a constant, which is used once to store the value in the constant and the next time it is not changed again.