QUIZ

Fill in the blanks in the following paragraph

In Go, a group of related variables can be defined as a `struct` . Each variable within a struct is known as a `field` .

You got it!

Which of the following defines an array of `Pet`s

```
pets := []Pet{{“Fido”, 10}, {“Jerry”, 3}, {“Stinky”, 4}, {“Jillip”, 2}}
```

Correct!

```
pets := {{“Fido”, 10}, {“Jerry”, 3}, {“Stinky”, 4}, {“Jillip”, 2}}
```

```
pets := []Pet{[“Fido”, 10], [“Jerry”, 3], [“Stinky”, 4], [“Jillip”, 2]}
```

```
pets := []Pet{new Pet{“Fido”, 10}, new Pet{“Jerry”, 3}, new Pet{“Stinky”, 4}, new Pet{“Jillip”, 2}}
```

Fill in the code to define an `Address` struct nested within a `Student` struct:

```
type Address struct {
        houseNumber int
        street string
}

type Student struct{
        name string
        gpa float32
        address Address
}
```

You got it!

Fill in the code below to instantiate an Item with a name of “Soda” and a price of 1.99.

```
item := Item {name: "Soda" price: 1.99}
```

You got it!

Fill in the code to make a `speak` method for a `Pet`.

```
func ( pet Pet ) speak() {
  fmt.Println(pet.name, "says:", pet.sound);
}
```

You got it!

Which of the following is a correct definition of a Pet struct with a name, and an age?

```
type Pet { name, age }
```

```
type Pet int
```

```
type Pet {}
```

```
type Pet struct{
        name string
        age int
}
```

Correct! A struct definition provides its name, field names, and field data types.

Which of the following is the syntax for accessing a field in a struct?

Square brackets `[]`

Curly braces `{}`

A dot `.`

 Correct!

An arrow `->`

Fill in the code to write a function which permanently updates the `Pet`'s age.

```
func (pet *Pet) birthday() {
    fmt.Println("Happy Birthday", pet.name)
    pet.age++
    fmt.Println(pet.name, "is now", pet.age)
}
```

 You got it!