

What is Experiment Driven Development?

Why is it most useful in statically typed languages?

Getting the hang of APIs or exploring module integration using an interactive session with the interpreter. This should come prior to writing tests and code.

Writing tests may be premature in a statically typed languages until you have experimented with the type signatures you will use.

How can one create a companion object in the REPL?

Put both the class and its companion inside some wrapper object.

What sorts of things are inexpressible in the  
REPL?

Certain aspects of coding with packages and visibility modifiers.

What's a shorthand for `AnyRef.hashCode`?

AnyRef.##



What's a good strategy for interacting with Java code that may return `null`?

Pass the possibly `null` reference to the `Option` factory method. It will return `None` or `Some` as appropriate, which can be pattern matched against.

What strategy can be used to minimize unnecessary pattern matching on `Option`?

Treating `Option` as a collection, using `for` expressions and methods like `foreach`, `map`, and `filter`.

What is the result of `for/yield` expressions over one or more `Options`?

Another Option of the same type.

What is the `scala.Equals` trait for?

Enforcing the existence of the `canEquals` method correct implementations of `equals` in polymorphic classes require.