Object Closures

Object closure: An object that plays the role of a function and its **referencing environment**.

annonymous method: An unnamed method.

The referencing environment in a closure will be nontrivial only when passing a nested subroutine. This mean that the implementation of first-class subroutines is trivial in a language without nested subroutines. Programming languages that do not allow nested subroutines are missing a useful feature: the ability to pass a subroutine *with context.* In object-oriented languages, there is an alternative way to achieve a similiar effect: we can encapsulate our subroutine as a method of a simple object, and let the object's fields hold context for the method.

An object that plays the role of a function and its referencing environment may be called an *object closure*, a *fuction object*, or a *functor*.