Maple allows you to group procedures together into a single object called a package. When you type:

```
with(LinearAlgebra)
```

all of the symbols in the LinearAlgebra package are loaded into the main namespace. You can access individual symbols in the package without loading it using the :- operator, e.g.

```
LinearAlgebra:-MatrixInverse
```

At some point you will want to write your own packages so that you can reuse code you have written. Here's one way.

A package is, technically, a Maple module with the option package specified.

```
Math412:=module()
option package;
# List symbols you want to export from the package here.
export UnitTangent;
# Define the symbols we want to export (and perhaps some symbols
# we want to keep private to the package.
UnitTangent:=proc(alpha,t)
local T, nT;
T:=simplify(map(diff,alpha,t)) assuming real;
nT:=simplify(sqrt(LinearAlgebra:-DotProduct(T,T))) assuming real;
T:=simplify(T/nT) assuming real;
return T;
end;
```

end;

If you make this definition in Maple, you can then access UnitTangent by either Math412:-UnitTangent or by loading the package first with with(Math412) and then using plain old UnitTangent. But when you quit Maple, the package will be gone.

To make a package that is persistent between Maple sessions, you need to store it in a file, known as a Maple repository. The command to do this is

```
LibraryTools:-Save(Math412,dirname)
```

where dirname is a path to the directory where you want to save the repository.

For this to be a useful operation, you must either save the repository to a location where Maple knows to look, or you need to tell Maple some new places to look for repositories. When you start Maple, Maple looks for repositories in

\$HOME\$/maple/toolbox/\$TOOLBOX_NAME\$/lib

where \$HOME\$ is your home directory and \$TOOLBOX_NAME\$ is any directory name. On my computer,

```
LibraryTools:-Save(Math412,"/Users/david/maple/toolbox/Math412/lib")
```

saves a repository to a location that Maple will find in the future.

Alternatively, you can adjust the Maple variable libname to tell Maple new places to look for repositories. See the Maple help for libname and .mapleinit if you want to go this route.

The Maple worksheet MakeAPackage.mw on the course web page contains an example of saving code in a package in a repository saved in a toolbox directory. If you run the worksheet, you will have saved a package called Math412 and you should be able to use Math412:-UnitTangent in future Maple sessions without doing any extra work.