

**Usage:** CreateMakefile.sh [ExecutableName] ( [cflags.txt] )

I'm using square brackets because I'm having a problem with the escape character, and I wasn't able to fix it. I don't want it to be a problem.

**Case 1:** no main file exists, no Makefile exists, one file, no cflags passed in

\$ ls

AddConsDes.sh	InitClassFiles.sh	getSuffix.sh
AddGetSetHeader.sh	InsertLibrary.sh	getValue.sh
AddGetSetSource.sh	IsLibraryPresent.sh	headerMakefile.txt
AddHeader.sh	Types.table	CreateMakefile.sh
AddMethod.sh	Valerie_861102819.txt	
listLibraries.txt	getLibrary.sh	
InheritanceTree.txt	getPrefix.sh	

\$ ./InitClassFiles.sh MyClass Member1:Integer Member2:Boolean

\$ ls

AddConsDes.sh	InitClassFiles.sh	getPrefix.sh
AddGetSetHeader.sh	InsertLibrary.sh	getSuffix.sh
AddGetSetSource.sh	IsLibraryPresent.sh	getValue.sh
AddHeader.sh	MyClass.cc	Types.table
AddMethod.sh	MyClass.hh	headerMakefile.txt
listLibraries.txt	Valerie_861102819.txt	<b>classList.txt</b>
InheritanceTree.txt	getLibrary.sh	CreateMakefile.sh

\$ ./CreateMakefile.sh myProgram

\$ cat Makefile

```
<header>
PGRM = myProgram
OBSJ = main.o MyClass.o
CC = g++
CFLAGS =

all: $(PGRM)

$(PGRM): $(OBSJ)
    $(CC) $(OBSJ) -o $(PGRM)

main.o: main.cc
    $(CC) $(CFLAGS) -c main.cc

MyClass.o: MyClass.cc
    $(CC) $(CFLAGS) -c MyClass.cc

clean:
    rm *.o $(PGRM) $(PGRM).tar.gz

tar:
    tar -czvf $(PGRM).tar.gz *.cc *.hh
```

\$ make all

```
g++ -c main.cc
g++ -c MyClass.cc
g++ main.o MyClass.o -o myProgram
```

\$ make MyClass.o

```
make: `MyClass.o' is up to date.
```

\$ make tar

```
tar -czvf myProgram.tar.gz *.cc *.hh
a MyClass.cc
a main.cc
a MyClass.hh
```

\$ make clean

```
rm *.o myProgram myProgram.tar.gz
```

Notes:

- An empty main file `main.cc` is created if it does not already exist.
- The makefile `Makefile` is created if it does not already exist.
  - o If it exists, overwrite it
- A list of the classes created by `InitClassFiles.sh` ("`classList.txt`") is created if it does not already exist.

## Case 2: main and Makefile already exist, 2+ files, cflags passed in

(continuing off previous example)

```
$ ./InitClassFiles.sh AnotherClass Member3:Decimal Member4:Character
```

```
$ echo "-W -Wall -Werror" > cflags.txt
```

```
$ ls:
```

AddConsDes.sh	InitClassFiles.sh	getLibrary.sh	CreateMakefile.sh
AddGetSetHeader.sh	InsertLibrary.sh	getPrefix.sh	
AddGetSetSource.sh	IsLibraryPresent.sh	getSuffix.sh	
AddHeader.sh	<b>MyClass.cc</b>	getValue.sh	
AddMethod.sh	<b>MyClass.hh</b>	listLibraries.txt	
<b>AnotherClass.cc</b>	Types.table	classList.txt	
<b>AnotherClass.hh</b>	Valerie_861102819.txt	main.cc	
InheritanceTree.txt	headerMakefile.txt	<b>cflags.txt</b>	

```
$ ./CreateMakefile.sh myNextProgram cflags.txt
```

```
$ cat Makefile
```

```
<header>
PGRM = myNextProgram
OBSJ = main.o MyClass.o AnotherClass.o
CC = g++
CFLAGS = -W -Wall -Werror

all: $(PGRM)

$(PGRM): $(OBSJ)
    $(CC) $(OBSJ) -o $(PGRM)

main.o: main.cc
    $(CC) $(CFLAGS) -c main.cc

MyClass.o: MyClass.cc
    $(CC) $(CFLAGS) -c MyClass.cc

AnotherClass.o: AnotherClass.cc
    $(CC) $(CFLAGS) -c AnotherClass.cc

clean:
    rm *.o $(PGRM) $(PGRM).tar.gz

tar:
    tar -czvf $(PGRM).tar.gz *.cc
```

```
$ make all
```

g++ -W -Wall -Werror -c main.cc
g++ -W -Wall -Werror -c MyClass.cc
g++ -W -Wall -Werror -c AnotherClass.cc
g++ main.o MyClass.o AnotherClass.o -o myProgram

Modify AnotherClass.cc without errors (ie change whitespace)

\$ make AnotherClass.o

```
g++ -W -Wall -Werror -c AnotherClass.cc
```

\$ make tar –identical to make tar in Case 1 example

\$ make clean –identical to make clean in Case 1 example

Notes:

- Make sure to remove cflags.txt (if file exists) at the end of the script. (see Case 3)
- If user passes in cflags.txt without creating it, output error message and create Makefile with no flags.
- Check if \$2 is the valid cflags.txt file. If not, output error message and create Makefile with no flags.

**Case 3:** Correct dependencies in Makefile (a class includes another class, remove a class, misplaced files in same directory), and cflags not initialized (to make sure cflags.txt handled correctly in Case 2)  
(continuing off previous examples)

```
$ ./InitClassFiles.sh Point X:Decimal Y:Decimal
$ ./InitClassFiles.sh newClass Member1:Integer Member2:Boolean
$ ./AddMethod.sh newClass newMethod constant void Param1:Point Param2:Text
    notice: both Point and Text require their respective libraries
$ ls:
```

AddConsDes.sh	IsLibraryPresent.sh	getPrefix.sh
AddGetSetHeader.sh	Makefile	getSuffix.sh
AddGetSetSource.sh	MyClass.cc	getValue.sh
AddHeader.sh	MyClass.hh	headerMakefile.txt
AddMethod.sh	<b>Point.cc</b>	listLibraries.txt
AnotherClass.cc	<b>Point.hh</b>	main.cc
AnotherClass.hh	System_Tests.pdf	<b>newClass.cc</b>
CreateMakefile.sh	Types.table	<b>newClass.hh</b>
InheritanceTree.txt	Valerie_861102819.txt	test.sh
InitClassFiles.sh	classList.txt	
InsertLibrary.sh	getLibrary.sh	

```
$ ./CreateMakefile.sh finalProgram cflags.txt
```

```
Error: cflags.txt does not exist. Makefile will be generated with no cflags.
```

```
$ cat Makefile
```

```
<header>
PRGM = finalProgram
OBSJ = main.o AnotherClass.o Point.o newClass.o
CC = g++
CFLAGS =

all: $(PRGM)

$(PRGM): $(OBSJ)
    $(CC) $(OBSJ) -o $(PRGM)

main.o: main.cc
    $(CC) $(CFLAGS) -c main.cc

AnotherClass.o: AnotherClass.cc
    $(CC) $(CFLAGS) -c AnotherClass.cc

Point.o: Point.cc
    $(CC) $(CFLAGS) -c Point.cc

newClass.o: newClass.cc
    $(CC) $(CFLAGS) -c newClass.cc

clean:
    rm *.o $(PRGM) $(PRGM).tar.gz
```

```
$ make all
```

```
g++ -c main.cc
g++ -c MyClass.cc
g++ -c AnotherClass.cc
g++ -c Point.cc
g++ -c newClass.cc
g++ main.o MyClass.o AnotherClass.o Point.o newClass.o -o finalProgram
```

Make an error to AnotherClass.cc (ie syntax error)

```
$ make all
```

```
g++ -c AnotherClass.cc
AnotherClass.cc:32:1: error: unknown type name 'example'
example error
^
AnotherClass.cc:32:14: error: expected ';' after top
level declarator
example error
      ^
      ;
AnotherClass.cc:34:28: error: expected unqualified-id
AnotherClass::AnotherClass()
                        ^
3 errors generated.
make: *** [AnotherClass.o] Error 1
```

```
$ make tar - identical to make tar in Case 1&2 example
```

```
$ make clean - identical to make clean in Case 1&2 example
```

below is to make sure that unrelated files are not included

```
$ touch hello.cc hello.hh hello.o
```

```
$ ls
```

AddConsDes.sh	Makefile	getValue.sh
AddGetSetHeader.sh	MyClass.cc	headerMakefile.txt
AddGetSetSource.sh	MyClass.hh	<b>hello.cc</b>
AddHeader.sh	Point.cc	<b>hello.hh</b>
AddMethod.sh	Point.hh	<b>hello.o</b>
AnotherClass.cc	System_Tests.pdf	listLibraries.txt
AnotherClass.hh	Types.table	main.cc
CreateMakefile.sh	Valerie_861102819.txt	newClass.cc
InheritanceTree.txt	classList.txt	newClass.hh
InitClassFiles.sh	getLibrary.sh	test.sh
InsertLibrary.sh	getPrefix.sh	
IsLibraryPresent.sh	getSuffix.sh	

```
$ ./CreateMakefile.sh extraFileProgram
```

```
Attention: Files that do not belong to the program are present.
Please check your files, and move/delete them accordingly.
```

```
$ cat Makefile
```

```
PGRM = extraFileProgram
OBSJS = main.o MyClass.o AnotherClass.o Point.o newClass.o
CC = g++
CFLAGS =

all: $(PGRM)

$(PGRM): $(OBSJS)
    $(CC) $(OBSJS) -o $(PGRM)

main.o: main.cc
    $(CC) $(CFLAGS) -c main.cc

MyClass.o: MyClass.cc
    $(CC) $(CFLAGS) -c MyClass.cc

AnotherClass.o: AnotherClass.cc
    $(CC) $(CFLAGS) -c AnotherClass.cc

Point.o: Point.cc
    $(CC) $(CFLAGS) -c Point.cc

newClass.o: newClass.cc
    $(CC) $(CFLAGS) -c newClass.cc

clean:
    rm main.o MyClass.o AnotherClass.o Point.o newClass.o $(PGRM)$(PGRM).tar.gz

tar:
    tar -czvf $(PGRM).tar.gz main.cc MyClass.hh MyClass.cc AnotherClass.hh
    AnotherClass.cc Point.hh Point.cc newClass.hh newClass.cc
```

```
$ make all
```

```
g++ -c main.cc
g++ -c MyClass.cc
g++ -c AnotherClass.cc
g++ -c Point.cc
g++ -c newClass.cc
g++ main.o MyClass.o AnotherClass.o Point.o newClass.o -o finalProgram
```

```
$ make tar → tars successfully
```

```
$ make clean → clears the correct files
```