

```
1. //
2. // main.c
3. // HelloObjC
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Says hello to the world in Objective-C.
10. //
11.
12. #import <Foundation/Foundation.h>
13.
14. int main(int argc, const char * argv[])
15. {
16.     @autoreleasepool {
17.         NSLog(@"Hello, World!");
18.     }
19.     return 0;
20. }
```

```
1. //
2. // main.m
3. // Students1
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates use of a class.
10. //
11.
12. #import <Foundation/Foundation.h>
13. #import "Student.h"
14.
15. // prototype
16. void greet(Student *s);
17.
18. int main(int argc, const char * argv[])
19. {
20.     @autoreleasepool {
21.
22.         // Alice
23.         Student *alice = [Student alloc];
24.         alice->age = 20;
25.         alice->name = @"Alice";
26.         greet(alice);
27.
28.         // Bob
29.         Student *bob = [Student alloc];
30.         bob->age = 21;
31.         bob->name = @"Bob";
32.         greet(bob);
33.
34.     }
35.     return 0;
36. }
37.
38. // greets student (via stderr)
39. void greet(Student *s)
40. {
41.     NSLog(@"Hello, %@. I see that you are %d years old.\n", s->name, s->age);
42. }
```

```
1. //
2. // Student.h
3. // Students1
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Declares a student.
10. //
11.
12. #import <Foundation/Foundation.h>
13.
14.
15. @interface Student : NSObject {
16.     @public
17.         int age;
18.         NSString *name;
19. }
20. @end
```

```
1. //
2. // Student.m
3. // Students1
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Defines a student.
10. //
11.
12. #import "Student.h"
13.
14. @implementation Student
15. @end
```

```
1. //
2. // main.m
3. // Students2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates use of a class, getters, and setters.
10. //
11.
12. #import <Foundation/Foundation.h>
13. #import "Student.h"
14.
15. // prototype
16. void greet(Student *s);
17.
18. int main(int argc, const char * argv[])
19. {
20.     @autoreleasepool {
21.
22.         // Alice
23.         Student *alice = [Student alloc];
24.         [alice setAge:20];
25.         [alice setName:@"Alice"];
26.         greet(alice);
27.
28.         // Bob
29.         Student *bob = [Student alloc];
30.         [bob setAge:21];
31.         [bob setName:@"Bob"];
32.         greet(bob);
33.     }
34.     return 0;
35. }
36.
37. // greets student (via stderr)
38. void greet(Student *s)
39. {
40.     NSLog(@"Hello, %@. I see that you are %d years old.\n", [s name], [s age]);
41. }
```

```
1. //
2. // Student.h
3. // Students2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Declares a student with getters and setters.
10. //
11.
12. #import <Foundation/Foundation.h>
13.
14. @interface Student : NSObject {
15.     int _age;
16.     NSString *_name;
17. }
18.
19. - (int)age;
20. - (void)setAge:(int)age;
21.
22. - (NSString *)name;
23. - (void)setName:(NSString *)name;
24.
25. @end
```

```
1. //
2. // Student.m
3. // Students2
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Defines a student with getters and setters.
10. //
11.
12. #import "Student.h"
13.
14. @implementation Student
15.
16. - (int)age
17. {
18.     return _age;
19. }
20.
21. - (void)setAge:(int)age
22. {
23.     _age = age;
24. }
25.
26. - (NSString *)name
27. {
28.     return _name;
29. }
30.
31. - (void)setName:(NSString *)name
32. {
33.     _name = [name copy];
34. }
35.
36. @end
```

```
1. //
2. // main.m
3. // Students3
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates use of properties.
10. //
11.
12. #import <Foundation/Foundation.h>
13. #import "Student.h"
14.
15. // prototype
16. void greet(Student *s);
17.
18. int main(int argc, const char * argv[])
19. {
20.     @autoreleasepool {
21.
22.         // Alice
23.         Student *alice = [Student alloc];
24.         alice.age = 20;
25.         alice.name = @"Alice";
26.         greet(alice);
27.
28.         // Bob
29.         Student *bob = [Student alloc];
30.         bob.age = 21;
31.         bob.name = @"Bob";
32.         greet(bob);
33.     }
34.     return 0;
35. }
36.
37. // greets student (via stderr)
38. void greet(Student *s)
39. {
40.     NSLog(@"Hello, %@. I see that you are %d years old.\n", s.name, s.age);
41. }
```



```
1. //
2. // Student.h
3. // Students3
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Declares a student with properties.
10. //
11.
12. #import <Foundation/Foundation.h>
13.
14. @interface Student : NSObject {
15.     int _age;
16.     NSString *_name;
17. }
18.
19. @property (assign, nonatomic, readwrite) int age;
20. @property (copy, nonatomic, readwrite) NSString *name;
21.
22. - (int)age;
23. - (void)setAge:(int)age;
24.
25. - (NSString *)name;
26. - (void)setName:(NSString *)name;
27.
28. @end
```

```
1. //
2. // Student.m
3. // Students3
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Defines a student with properties.
10. //
11.
12. #import "Student.h"
13.
14. @implementation Student
15.
16. - (int)age
17. {
18.     return _age;
19. }
20.
21. - (void)setAge:(int)age
22. {
23.     _age = age;
24. }
25.
26. - (NSString *)name
27. {
28.     return _name;
29. }
30.
31. - (void)setName:(NSString *)name
32. {
33.     _name = [name copy];
34. }
35.
36. @end
```

```
1. //
2. // main.m
3. // Students4
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates use of synthesized properties.
10. //
11.
12. #import <Foundation/Foundation.h>
13. #import "Student.h"
14.
15. // prototype
16. void greet(Student *s);
17.
18. int main(int argc, const char * argv[])
19. {
20.     @autoreleasepool {
21.
22.         // Alice
23.         Student *alice = [Student alloc];
24.         alice.age = 20;
25.         alice.name = @"Alice";
26.         greet(alice);
27.
28.         // Bob
29.         Student *bob = [Student alloc];
30.         bob.age = 21;
31.         bob.name = @"Bob";
32.         greet(bob);
33.     }
34.     return 0;
35. }
36.
37. // greets student (via stderr)
38. void greet(Student *s)
39. {
40.     NSLog(@"Hello, %@. I see that you are %d years old.\n", s.name, s.age);
41. }
```

```
1. //
2. // Student.h
3. // Students4
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Declares a student with synthesized properties.
10. //
11.
12. #import <Foundation/Foundation.h>
13.
14. @interface Student : NSObject {
15. }
16.
17. @property (assign, nonatomic, readwrite) int age;
18. @property (copy, nonatomic, readwrite) NSString *name;
19.
20. @end
```

```
1. //
2. // Student.m
3. // Students4
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Defines a student with synthesized properties.
10. //
11.
12. #import "Student.h"
13.
14. @implementation Student
15.
16. @synthesize age=_age;
17. @synthesize name=_name;
18.
19. @end
```

```
1. //
2. // main.m
3. // Students5
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates use of (mostly) synthesized properties.
10. //
11.
12. #import <Foundation/Foundation.h>
13. #import "Student.h"
14.
15. // prototype
16. void greet(Student *s);
17.
18. int main(int argc, const char * argv[])
19. {
20.     @autoreleasepool {
21.
22.         // Alice
23.         Student *alice = [Student alloc];
24.         alice.age = 20;
25.         alice.name = @"Alice";
26.         greet(alice);
27.
28.         // Bob
29.         Student *bob = [Student alloc];
30.         bob.age = 21;
31.         bob.name = @"Bob";
32.         greet(bob);
33.
34.         // David
35.         Student *david = [Student alloc];
36.         david.age = 34;
37.         david.name = @"David";
38.         greet(david);
39.     }
40.     return 0;
41. }
42.
43. // greets student (via stderr)
44. void greet(Student *s)
45. {
46.     NSLog(@"Hello, %@. I see that you are %d years old.\n", s.name, s.age);
47. }
```

```
1. //
2. // Student.h
3. // Students5
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Declares a student with (mostly) synthesized properties.
10. //
11.
12. #import <Foundation/Foundation.h>
13.
14. @interface Student : NSObject {
15. }
16.
17. @property (assign, nonatomic, readwrite) int age;
18. @property (copy, nonatomic, readwrite) NSString *name;
19.
20. @end
```

```
1. //
2. // Student.m
3. // Students5
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Defines a student with (mostly) synthesized properties.
10. //
11.
12. #import "Student.h"
13.
14. @implementation Student
15.
16. @synthesize age=_age;
17. @synthesize name=_name;
18.
19. - (void)setName:(NSString *)name
20. {
21.     if ([name isEqualToString:@"David"]) {
22.         _name = [[NSString alloc] initWithString:@"Dummy"];
23.     }
24.     else {
25.         _name = [name copy];
26.     }
27. }
28.
29. @end
```



```
1. //
2. // main.m
3. // Students6
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates init methods.
10. //
11.
12. #import <Foundation/Foundation.h>
13. #import "Student.h"
14.
15. // prototype
16. void greet(Student *s);
17.
18. int main(int argc, const char * argv[])
19. {
20.     @autoreleasepool {
21.
22.         // Alice
23.         Student *alice = [[Student alloc] initWithName:@"Alice" andAge:20];
24.         greet(alice);
25.
26.         // Bob
27.         Student *bob = [[Student alloc] initWithName:@"Bob" andAge:21];
28.         greet(bob);
29.
30.         // John
31.         Student *john = [[Student alloc] init];
32.         greet(john);
33.     }
34.     return 0;
35. }
36.
37. // greets student (via stderr)
38. void greet(Student *s)
39. {
40.     NSLog(@"Hello, %@. I see that you are %d years old.\n", s.name, s.age);
41. }
```

```
1. //
2. // Student.h
3. // Students6
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Declares a student with init methods.
10. //
11.
12. #import <Foundation/Foundation.h>
13.
14. @interface Student : NSObject {
15. }
16.
17. @property (assign, nonatomic, readwrite) int age;
18. @property (copy, nonatomic, readwrite) NSString *name;
19.
20. - (id)initWithName:(NSString *)name andAge:(int)age;
21.
22. @end
```

```
1. //
2. // Student.m
3. // Students6
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Defines a student with init methods.
10. //
11.
12. #import "Student.h"
13.
14. @implementation Student
15.
16. @synthesize age=_age;
17. @synthesize name=_name;
18.
19. - (id)init
20. {
21.     self = [self initWithName:@"John" andAge:404];
22.     return self;
23. }
24.
25. - (id)initWithName:(NSString *)name andAge:(int)age
26. {
27.     if (self = [super init])
28.     {
29.         self.age = age;
30.         self.name = name;
31.     }
32.     return self;
33. }
34.
35. @end
```

```
1. //
2. // main.m
3. // Students7
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Demonstrates mutable arrays.
10. //
11.
12. #import <Foundation/Foundation.h>
13. #import "Student.h"
14.
15. // prototype
16. void greet(Student *s);
17.
18. int main(int argc, const char * argv[])
19. {
20.     @autoreleasepool {
21.
22.         // allocate array for students
23.         NSMutableArray *students = [[NSMutableArray alloc] init];
24.
25.         // Alice
26.         [students addObject:[[Student alloc] initWithName:@"Alice" andAge:20]];
27.
28.         // Bob
29.         [students addObject:[[Student alloc] initWithName:@"Bob" andAge:21]];
30.
31.         // greet then release each student
32.         for (Student *s in students) {
33.             greet(s);
34.         }
35.     }
36.     return 0;
37. }
38.
39. // greets student (via stderr)
40. void greet(Student *s)
41. {
42.     NSLog(@"Hello, %@. I see that you are %d years old.\n", s.name, s.age);
43. }
```

```
1. //
2. // Student.h
3. // Students7
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Declares a student with init methods.
10. //
11.
12. #import <Foundation/Foundation.h>
13.
14. @interface Student : NSObject {
15. }
16.
17. @property (assign, nonatomic, readwrite) int age;
18. @property (copy, nonatomic, readwrite) NSString *name;
19.
20. - (id)initWithName:(NSString *)name andAge:(int)age;
21.
22. @end
```

```
1. //
2. // Student.m
3. // Students7
4. //
5. // David J. Malan
6. // Harvard University
7. // malan@harvard.edu
8. //
9. // Defines a student with init methods.
10. //
11.
12. #import "Student.h"
13.
14. @implementation Student
15.
16. @synthesize age=_age;
17. @synthesize name=_name;
18.
19. - (id)init
20. {
21.     self = [self initWithName:@"John" andAge:404];
22.     return self;
23. }
24.
25. - (id)initWithName:(NSString *)name andAge:(int)age
26. {
27.     if (self = [super init])
28.     {
29.         self.age = age;
30.         self.name = name;
31.     }
32.     return self;
33. }
34.
35. @end
```