iOS

Objective-C, Window-based Applications, MVC

Objective-C

```objc
#import <Foundation/Foundation.h>

int main(int argc, const char * argv[])
{
    NSAutoreleasePool * pool = [[NSAutoreleasePool alloc] init];
    NSLog(@"Hello, World!");
    [pool drain];
    return 0;
}
```

Data Types

```
BOOL

id

nil

...
```

Foundation Data Types

NSInteger

NSPoint

NSRect

NSSize

NSUInteger

...

.h

```
@interface Foo: NSObject {
    // instance variables
}
// declarations of methods
@end
```

.m

```
@implementation Foo
// definitions of methods
@end
```

Instance Variables

```
@protected

@private

@public
```

Class Methods

```
+alloc;
```

Instance Methods

```
- (int)age;

- (void)setAge:(int)age;


- (void)init;

- (void)initWithName:(NSString *)name andAge:(int)age;
```

Messages

```
[student age];
[student setAge:20];


[student init];
[student initWithName:@"Alice" andAge:20];
```

Selectors

```
alloc

age
setAge:

init
initWithName:andAge:
```

@property

assign
copy
retain

atomic
nonatomic

readonly
readwrite

@synthesize

Memory Management

alloc   autorelease   dealloc   release   retain

Collections

```
NSArray
NSMutableArray

NSDictionary
NSMutableDictionary

NSSet
NSMutableSet

...
```

Fast Enumeration

```
for (id foo in bar) {
    // do something with foo
}
```

Categories

```
@interface Foo (Bar)

-(void)baz;

@end
```

Protocols

```
@interface Student <NSCopying> {
}

...

@end
```

Protocols

```objc
@implementation Student

...

-(id)copyWithZone:(NSZone *)zone
{
    Student *s = [Student allocWithZone:zone];
    [s initWithName:_name andAge:_age];
    return s;
}

...

@end
```

NSException

```
@try {

    // try something here

}
@catch (NSException *e) {

    // handle exception here

}
@finally {

    // do something here

}
```

NSError

```
NSError *e = nil;
if ([foo bar:baz error:&e] == nil)
{
    // handle error
}
```

MVC