Navigator,DocumentOutline(뷰컨에대응하는뷰의모든요소계층표시),Output/Debugger, Inspector,editor/Navigator:프로젝트파일,테스트등프로젝트에포함되는입력,디버깅정보,버전컨트롤/Editor:소스코드,스토리보드를이용한UI에디팅/Inspector:Editor에서 선택된 오브젝트에 대한 속성 등/Output:실행 시 발생하는 출력,브레이크 포인트에서 각종 변수 등의 값/Model-View-Controller은 iOS개발에 사용되는 핵심 디자인 패턴/:MVC에서 각 객체는 모델,뷰,컨트롤러 레이어 중의 하나/모델레이어:데이터를 저장하지만 UI에 대하여는 모름, simpleApp앱에서 모델은 문제와 정답의 목록을 가지는 문자열 리스트로구성/대게 모델 레이어는 현실세계의 사물을 대상/뷰레이어:사용자에게 보이는 모든 객체, 버튼,텍스트필드,슬라이더 등,대부분 이미 만들어진 것 그대로 사용/컨트롤러레이어:응용프로그램 관리, 사용자에게 보이는 뷰를 구성, 뷰와 모델 연결및동기화/뷰와모델은직접적으로통신안함,항상분리가능/사용자입력(뷰객체와상호작용)->뷰->컨트롤러에게메시지->컨은모델생신->컨은뷰가필요한모델객체에서데이터가져옴->컨은모델객체변경으로뷰갱신/IOS13multiScene지원,단말기커지는거대비/앱델리게이트->여러씬델리/모델->뷰컨->뷰/컨트롤러:뷰컨트롤러/모델:string인스턴스/뷰:라벨,버튼/storyboard:안드xml동일,뷰를디자인/,하나의스토리보드에여러뷰컨에해당하는뷰들어감/ObjectLibrary열고D&D하면View에표시/SB와UIVC동시오픈,SB에원하는컨트롤을Control+드래그로 UIVC클래스내에드롭->Outlet/Action선택,적절한Name지정/Outlet:객체의레퍼런스/Action:객체의listner,Type지정(UIButton)/동그라미생기는데이거는컨트롤과연결되었음을의미,비어있으면에러/Connection Inspector(인스팩터마지막탭)에서 연결확인가능,x누르면연결끊/viewDidLoad():프로그램완전실행직후처음호출되는callback함수,view초기화,생성자는아님/기기별아이콘전부준비(여러기기적용하기위해)/앱로딩화면LaunchScreen.storyBoard/General-App Icons & Launch Images -Launch Screen File:LaunchScreen선택/론칭시간제어Thread.sleep(fortimeInterval:2.0):2초/swift는애플의ios와macOS를위한프언20140602WWDC(애플세계개발자회의),기존오브젝티브-C와공존하기위해,클로저,다중리턴타입,네임스페이스,제네릭스,타입유추등오-C에없던현대프언기능포함,현재5.0/철학:오소,safe(안정성,엄격한문법실수방지,옵셔널quard,오류처리,타입통제),fast(신속성,c,c++,Objective-c대체),expressive(표현성:세미콜제거,사용편하고보기좋게),환경:Xcode(playground:swift해석기,interpreter처럼작동하지만swift는컴파일언어,xcode에내장,online있),REPL(Read-eval-print-loop)/var변수let상수/명시적타입let I :Int =70암시는rvalue로추론/변수위option+click으로타입확인/형변환:좌우동일해야,생성자호출로변환,Int(“10”)String(width)/

문자열로변환\(apples+oranges)/배열명시선언var list1 : [Int] = [] 암시var list2 = [Int]()/var list3 = [“”,””] 초기화암시list3.first/딕셔너리<key,value>로값저장,명시var dict : [String: Int] = [:],암시var dict = [String: Int]()/var dict = [“J” : “j”, “P”:”p”,] dict[“J”]=”JJ”/if a && b -> if a,b/switch는 break자동,default:…없으면case로다커버해야함/for in range: let range = 0..<10/0…10/for i in range{}/for \_ in 1…5{}/\_는변수생성x/for(k,ns)in dict {for n in ns}{}/repeat{i+1}while i < 10/func greet(name:String,day:String)->String{…}/func greet(from name:String, today day:String)->String{return”\(name)\(day)”}/print(greet(from:”Bob”, today:”Tuesday”))/순서동일/func s(ns:Int…)/func s(ns:[Int])/s(ns:4,1)/s(ns:[4,1])/중첩함수는감싼변수참조가능/함수도최상위타입:다른객체들에적용가능한연산을모두지원하는객체(상수변수등),함매넘,수,변대,같은연산지원시최상위타입객체,어떤함수가다른함수를반환값형태로반환가능/클로저:이름없는함수,중괄호,in으로인자와반환값타입분리/transform (T)->T배열리턴/에러시throw/ns.map({(n) in 3\*n})/매개변수번호:$0/ns.map({3\*$0})/new키워드없음/초기화(생성자)init(i:Int){…}/deinit파괴자/override/속성get{}set{}/protocol:자바interface/struct는메서드,이니셜등클래스와비슷한기능지원,값복사형태로전달,클은참조복사/extension:기존타입에새매서드,속성추가,기능재정의불가,연산프로퍼티,메소드,이니셜등추가,프로퍼티는안됨/optional:컴파일시값안정해진변수Int?/Int(“10”)!/Int(“10.0”)=nil/ImplicitUnWrappingOptional(Type!):일시nil가지기가능,read전반드시값저장안그럼죽음ForcedUnwrapping(강제언래핑)/if let a = b {a유효}/guard let … else {}/Optional Chaining nil이면하위접근안함nil반환,변수?/타입캐스팅as키워드/형확인is연산자/as?=nil,as!=죽음/다운캐스팅만캐스팅/try!에러없,try?에러nil/throws/뷰:UIView인스턴스or하위클래스인스턴스,통상단말뷰를컨트롤또는위젯/뷰는터치이벤트처리,뷰인스턴스는뷰계층구조상에존재,뷰계층구조의루는앱윈도우/모든앱은하나의UIWindow인스턴스가짐,여러뷰는여기포함/화면은CALayer에그린후전체에합성/cocoatouchclass로VC새로생성,스토리보드customclass에서 연결/init-loadView-ViewDidLoad-viewWillAppear-viewDidAppear-viewWillDisappear-viewDidDisappear(생성자-뷰Hierarchy완성,뷰가화면나오기전초기화,화면보이기직전(UIlayout조정),화면이보이는상태인터렉션,화면이안보이는시점코딩x,화면사라진후호출통상적으로저장될거잇으면저장)