

یک برنامه ساده با Swift

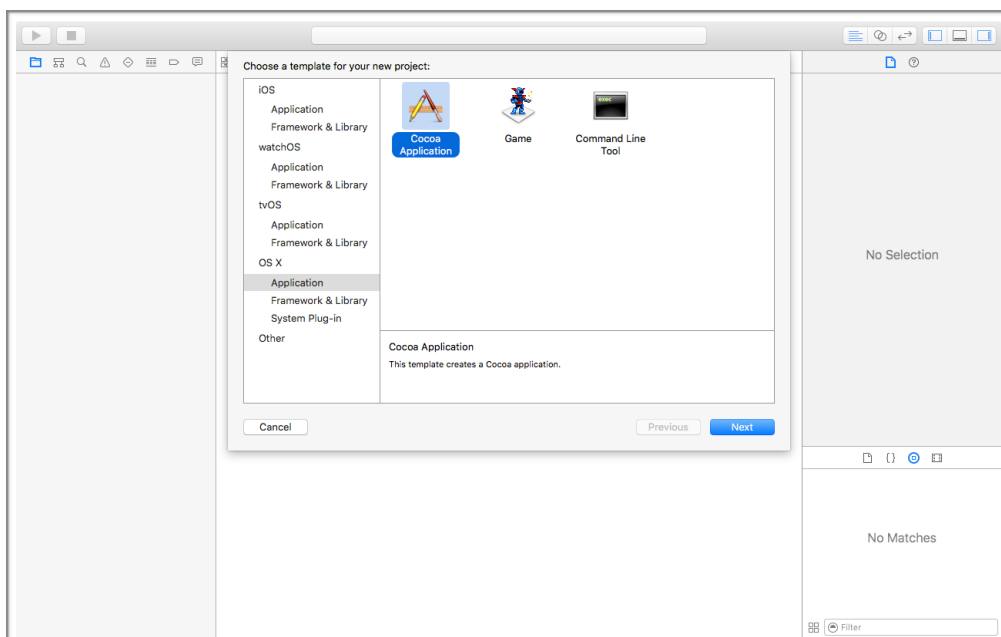
جمع دو عدد

برنامه برای مک

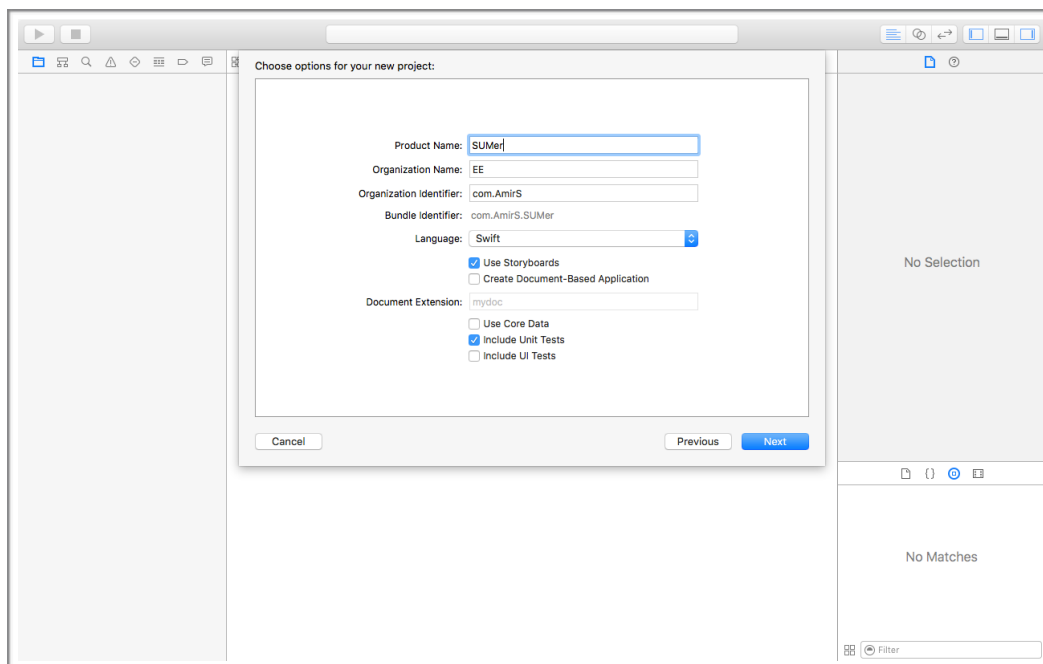
در این راهنمای کوتاه، قرار است برنامه‌ای ساده بسازیم، که دو عدد را دریافت و با هم جمع کند. البته همین برنامه ساده هم ریزه کاری‌های زیادی دارد، ولی تا حد امکان از آنها می‌گذریم و ابتدایی‌ترین کار ممکن را انجام خواهیم داد. ابتدا برنامه Xcode را باز کنید. اگر صفحه شروع باز شد، گزینه Create a new Xcode project را انتخاب کنید:



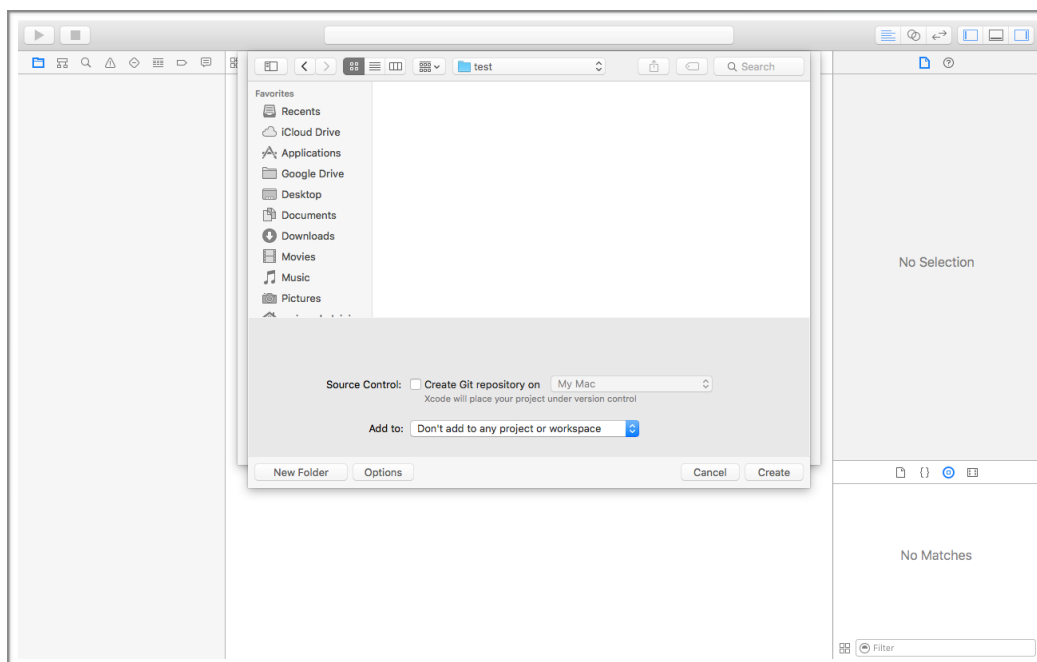
اگر این پنجره را بسته‌اید، همین گزینه در زیرمجموعه New در منوی File هم وجود دارد. بعد از باید نوع پروژه را از بین قالب‌های موجود انتخاب کنید تا Xcode متناسب با آن کارهای لازم را انجام دهد. چون قرار است برنامه را برای مک بسازید، در پنجره‌ای که باز شده، از لیست سمت چپ گزینه Application را از گروه OS X انتخاب کنید و بعد از بین آیکان‌های وسط کادر Cocoa Application را انتخاب کنید:



در صفحه بعد سه خانه اول را با هر اسمی که دوست دارید پر کنید و زبان را هم اگر Swift نیست، به Swift تغییر بدهید. با بقیه گزینه‌ها هم کاری نداشته باشید:



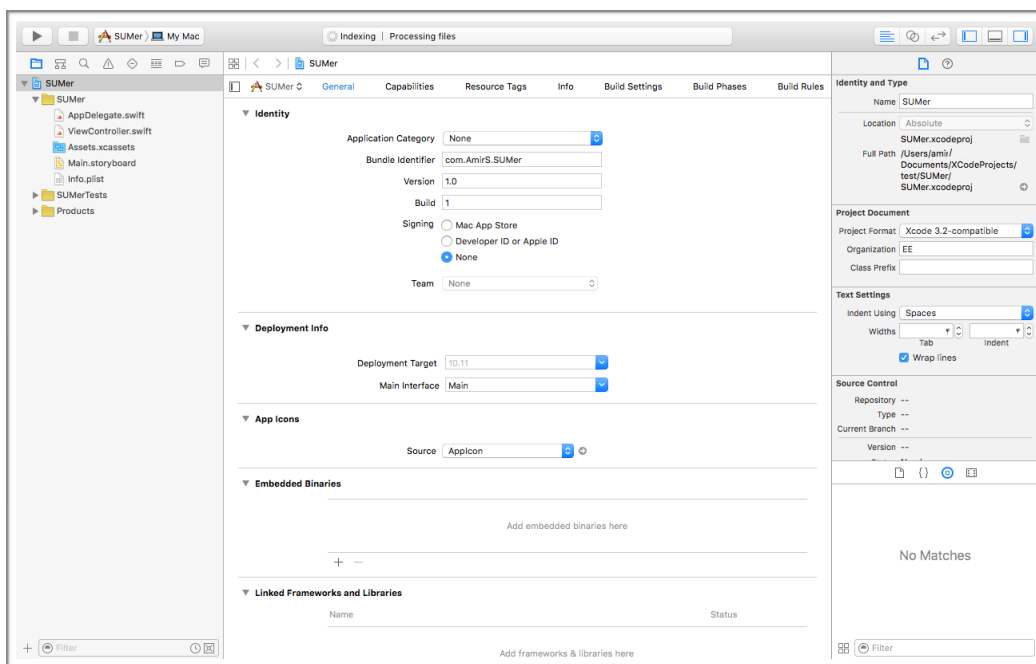
بعد، محل مناسبی را انتخاب کنید تا فایل‌های پروژه‌تان آنجا ذخیره شود:



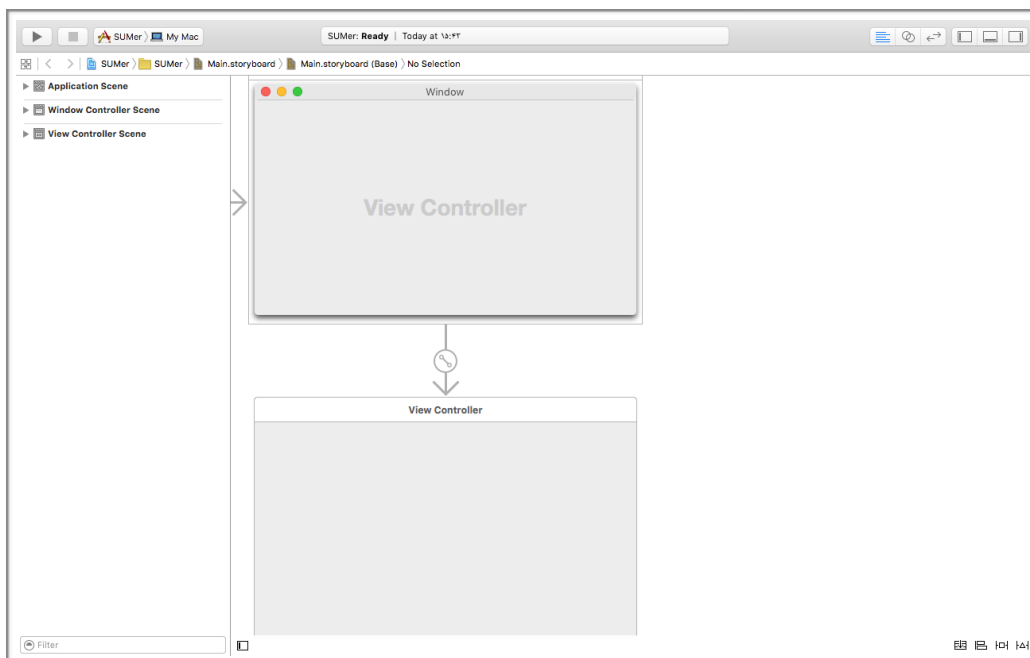
و به این ترتیب پروژه شما با حداقل‌های مورد نیاز آماده می‌شود. در پنجره‌ای که باز می‌شود، چهار قسمت اصلی وجود دارد: در نوار ابزار بالایی، در سمت راست، کلیدهایی برای باز و بسته کردن قسمت‌های دیگر قرار گرفته است. در سمت راست هم کلید جرا و قطع برنامه، مهم‌ترین چیزی است که با آن کار می‌خواهید کرد.

در قسمت اصلی، بخش وسط، چیزی که در حال کار روی آن هستید را می‌بینید، بخش سمت راست، متناسب با بخش مرکزی تغییر می‌کند و ابزارهای مورد نیاز را در اختیار می‌گذارد، بخش سمت چپ هم پنل‌های مختلفی دارد که در این پروژه فقط با درخت فایل‌های پروژه کار می‌کنید.

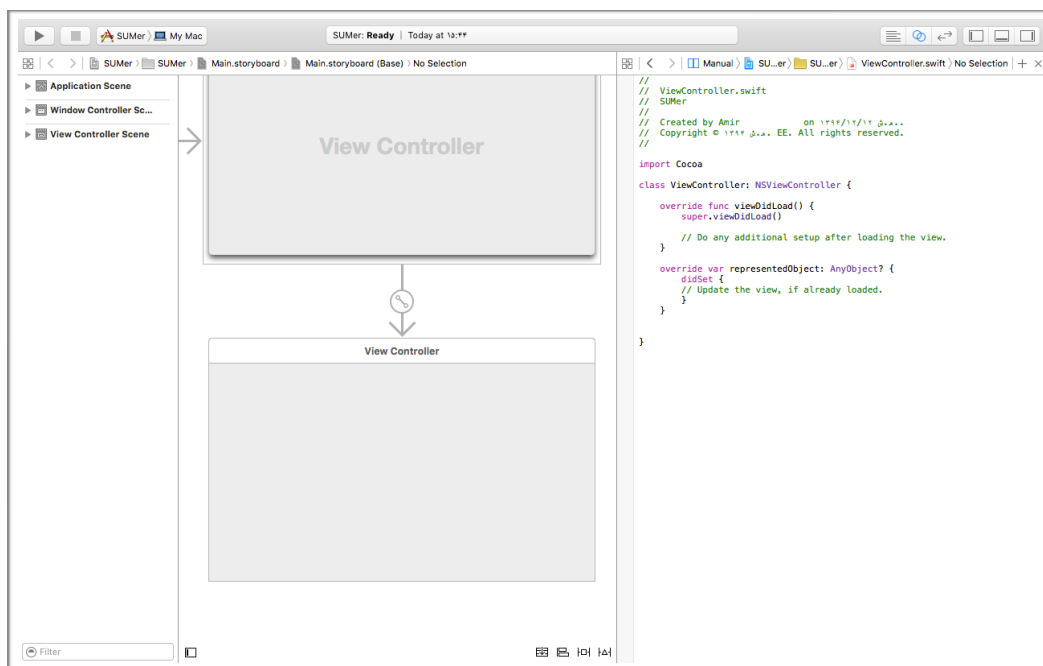
البته یک کادر پایینی هم وجود دارد که خروجی Log و چیزهای دیگری که مربوط به عیب‌یابی هستند، در آن قرار گرفته‌اند و فعلاً کاری به کارشان ندارید.



برای شروع، از لیست فایل‌ها، Main.storyboard را باز کنید. یادتان باشد که اگر روی چیزی دابل کلیک کنید، آن را در یک پنجره جدید باز می‌کند؛ این برای وقتی که می‌خواهید مرتب بین چندین فایل جابجا شوید به دردتان می‌خورد. فایلی که باز کرده‌اید، ظاهر رابط کاربری برنامه شما را می‌سازد. این برنامه فعلاً فقط یک پنجره تک صفحه‌ای دارد که آن را در تصویر می‌بینید:



یک فایل دیگر به نام ViewController.swift هم وجود دارد که کدهای مورد نیاز برای کار کردن برنامه را در آن خواهید نوشت. برای اینکه این دو فایل را کنار هم ببینید، روی ویرایشگر کمیکی که در سمت راست نوار ابزارها قرار دارد و به شکل دو دایره روی هم است، کلیک کنید. به این ترتیب فضای ویرایشگر به دو بخش تقسیم می‌شود. سمت چپ همچنان Main.storyboard را نشان می‌دهد و سمت راست هم به احتمال بسیار زیاد ViewController.swift را باز می‌کند؛ اما اگر چیز دیگری را نشان داد، نگران نشوید، کافی است در بهش بالای آن روی Automatic کلیک کنید و در منویی که باز می‌شود، ViewController.swift را پیدا و انتخاب کنید.

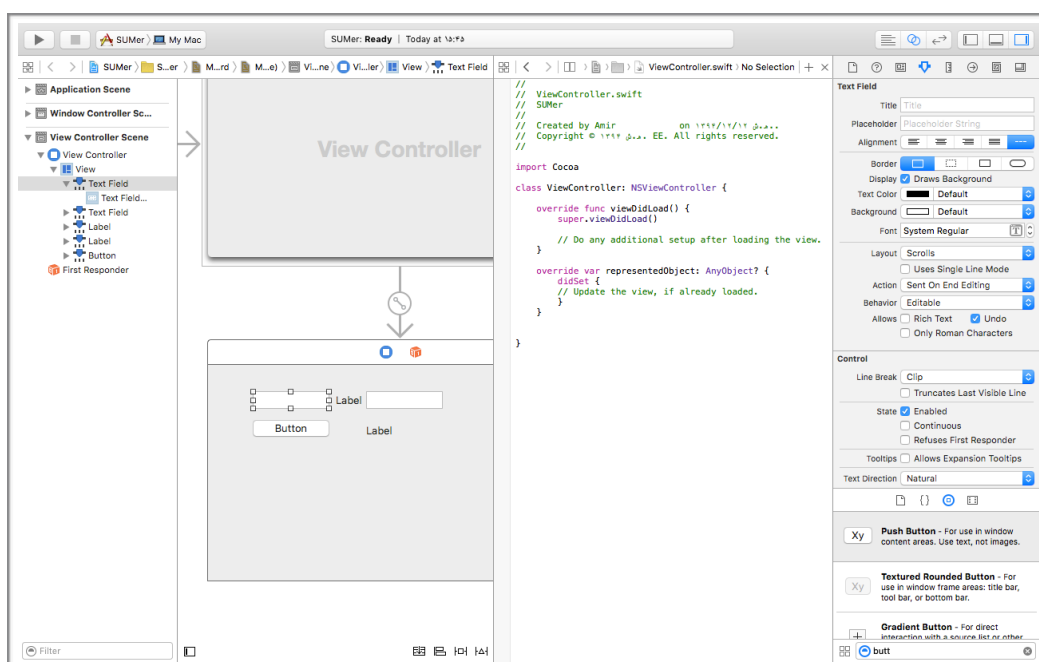


تا اینجا کار باید پنجره شما شبیه این تصویر شده باشد:

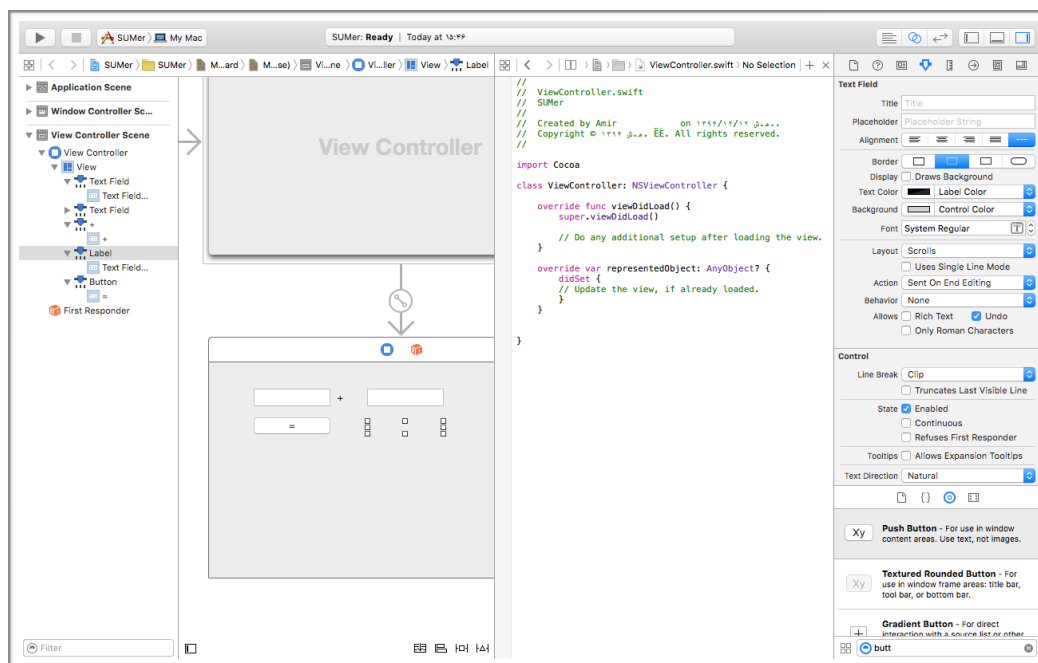
البته در این تصویر پنل‌های سمت راست و چپ بسته هستند؛ اما برای ساختن رابط کاربری، پنل سمت راست را نیاز دارید. پس آن را با کلیدش که در گوشه سمت راست نوار ابزار قرار گرفته، باز کنید.

اگر یادتان باشد، قرار بود برنامه‌تان دو عدد را بگیرد و با زدن یک دکمه آنها را جمع کرده و جواب را نمایش بدهد. به این ترتیب به دو Text Field نیاز دارید که اعداد اولیه در آن نوشته شود، یک Label برای نوشتن جواب لازم است، یک Button هم باید برای دستور جمع در نظر گرفته شود. علاوه بر اینها، بد نیست که یک Label هم برای نوشتن علامت جمع در نظر بگیرید تا معلوم شود که برنامه قرار است چکار کند.

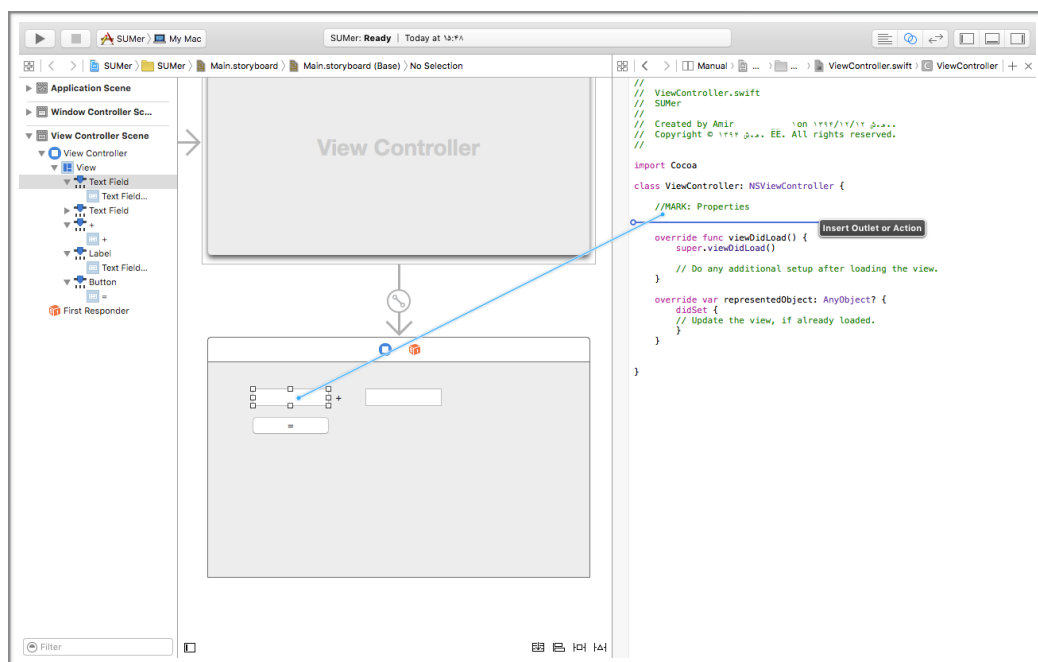
برای ساختن این اجزا، در پایین پنل سمت راست دکمه کتابخانه اشیا را که شبیه دکمه خانه در آیفون‌های قدیمی است را بزنید تا باز شود، بعد در کادر فیلتر که زیر آن قرار گرفته، چند حرف از این اسم‌هایی را که بالا به انگلیسی نوشتم، وارد کنید تا آن را برای شما پیدا کند. بعد آنها را داخل storyboard بکشید. در نهایت باید چیزی شبیه این تصویر ساخته باشید:



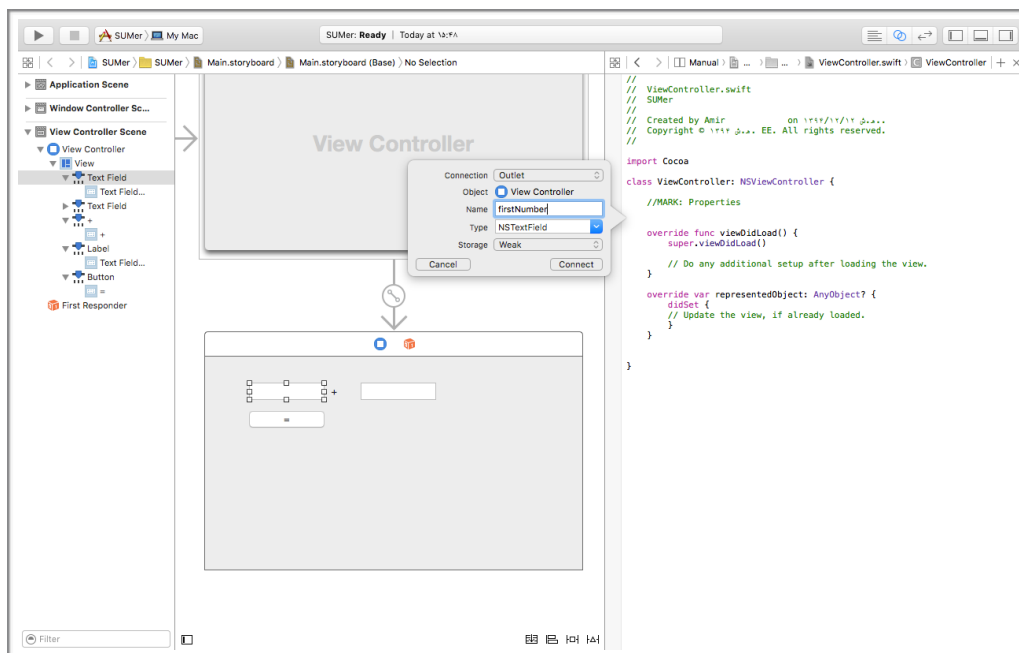
البته چیدن اجزای واسط کاربری و تنظیم اندازه و محل آنها، مخصوصا وقتی که اندازه پنجره تغییر کند، خیلی بیشتر کار می‌برد؛ حتی ممکن است چیزی لازم داشته باشید که Xcode بصورت آماده نداشته باشد و مجبور بشوید خودتان آن را بسازید. ولی فعلا با همین کنار بیایید. فقط بهتر است نوشته‌ها را تغییر دهید. برای اینکار روی آنها دابل‌کلیک کرده و هرچه می‌خواهید بنویسید. مثلا در تصویر زیر کادرهای اعداد اولیه و جواب جمع خالی است، در دکمه اجرا، علامت مساوی نوشته شده و علامت جمع هم بین دو عدد قرار داده شده است:



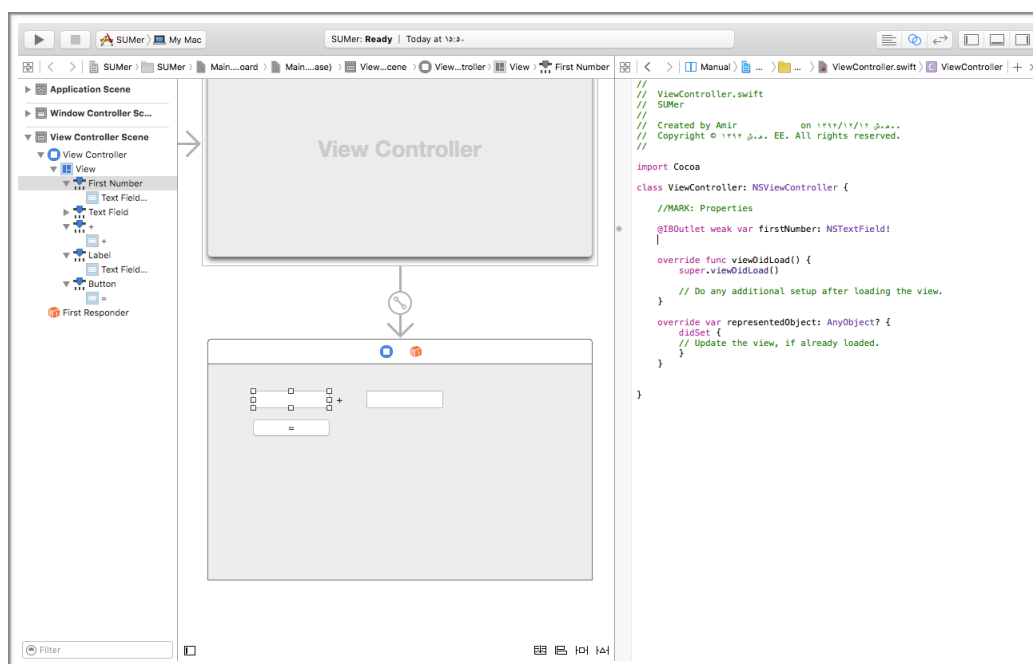
حالا باید اینها را به کد برنامه وصل کنید. برای اینکار کافی است کلید control را روی کیبورد نگه دارید و شی مورد نظر را به داخل کد بکشید و در محل مناسب رها کنید. تنها چیزی که باید رعایت کنید، این است که آن را در بلوک تعریف کلاس View Controller بگذارید و مواظب باشید که داخل بلوک‌های درون آن نروند. اما بد نیست که کارتان کمی نظم داشته باشد. مثلا می‌توانید محل مورد نظرتان را با نوشته MARK: // و اضافه کردن کلمه مناسب، بخش‌های کدتان را علامت‌گذاری کنید. مثلا در این تصویر بخشی به نام Properties برای کادرهای متن و لیبل درست شده است:



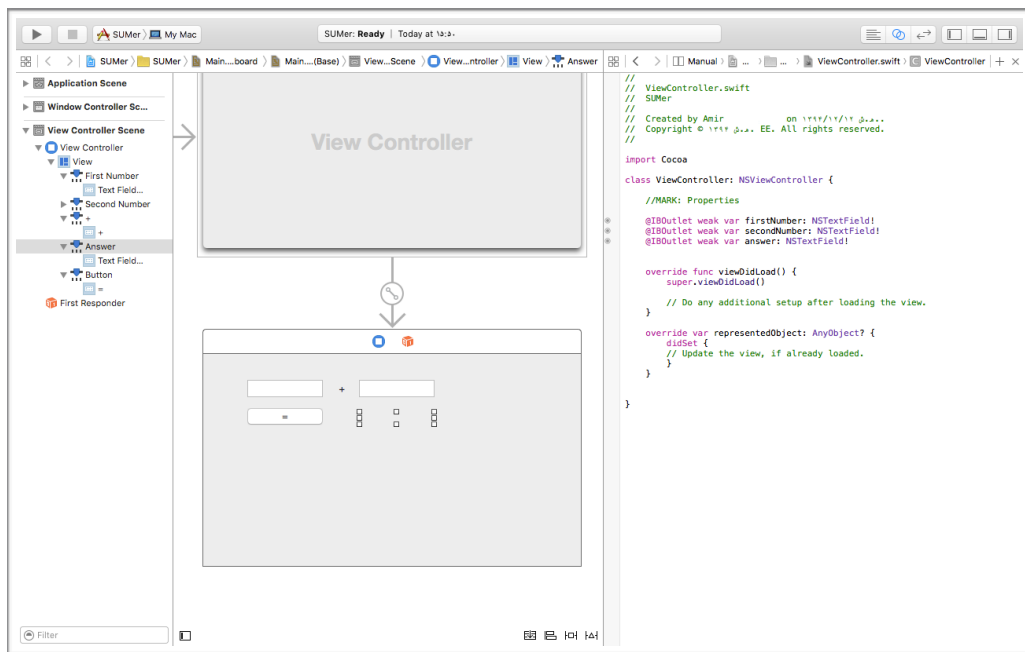
بعد از اینکه کلید ماوس را رها کردید، کادری باز می شود که باید در آن چیزهایی مثل نوع شی و اسم آن را وارد کنید. برای کادرهای متنی، نوع اتصال را Outlet بگذارید، اسم مناسبی بنویسید و بقیه را دست نزنید:



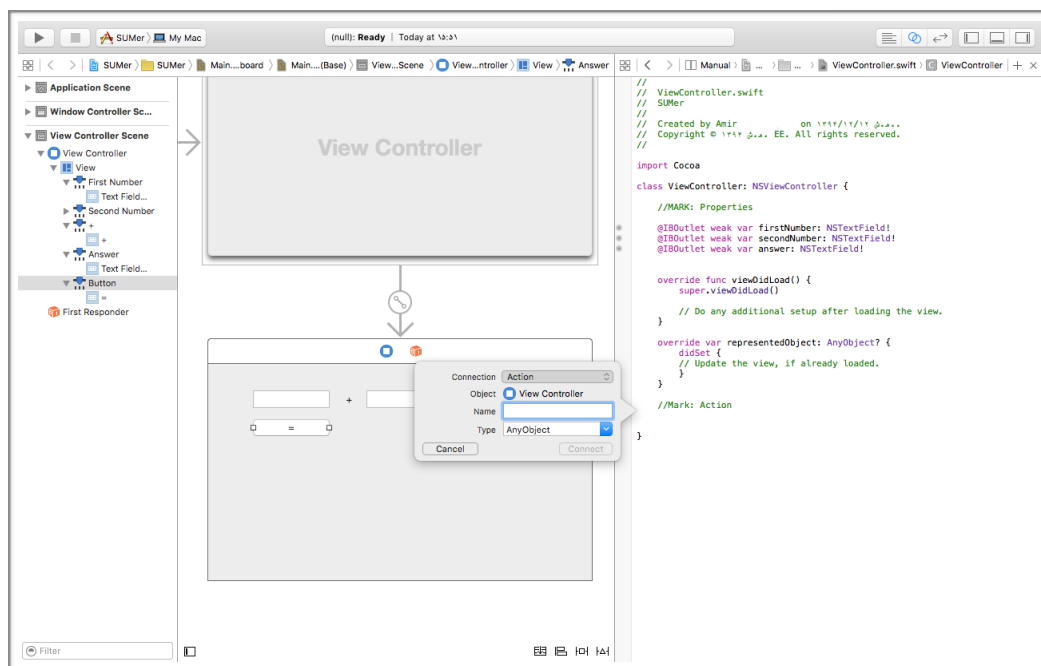
بعد کلید Connect را بزنید تا Xcode، کارهای لازم را انجام دهد و کد مورد نیاز را بنویسد:



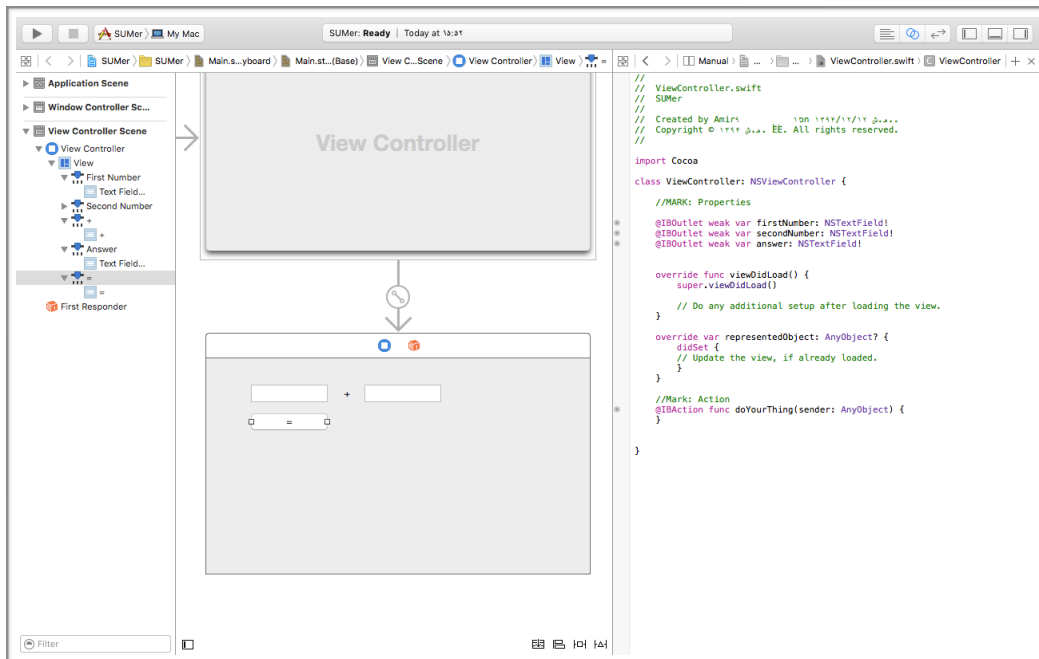
همین کار را، بجز دکمه مساوی، برای بقیه هم تکرار کنید. همانطور که در این تصویر هم می بینید، مرسوم است که اسم ها را با حروف کوچک بنویسند و فقط حرف اول کلمات دوم و بعد از آن را بزرگ بگذارند تا خواندنش ساده تر شود:



حالا کلید مساوی را هم وصل کنید، ولی دقت کنید که چون قرار است کاری انجام بدهد، باید نوع اتصال آن را Action بگذارید:

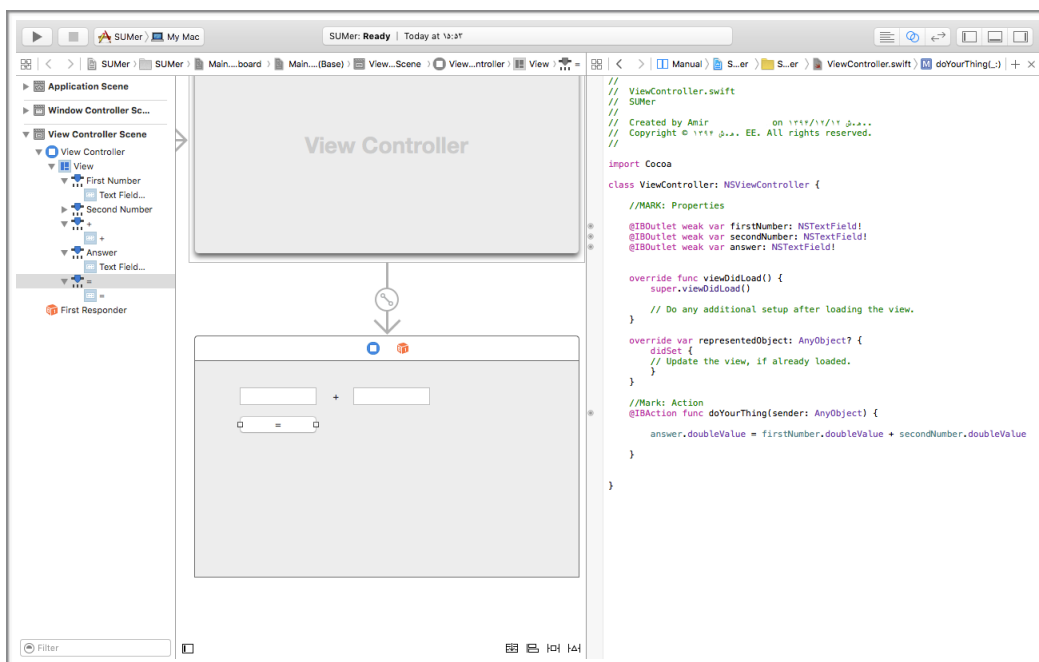


حالا کافی است بین آکولادهای اکشنی که ساخته شده است، دستوری بنویسید که مقدار موجود در دو کادر متن firstNumber و secondNumber را با هم جمع کرده و در answer قرار دهد. البته کار همیشه به این سادگی نیست، چون ممکن است چیزی که در این دو کادر نوشته می شود اصلا عدد نباشد یا ترکیبی از اعداد و حروف باشد یا حتی خالی بماند که در اینصورت باید برای چنین شرایطی هم عکس العمل مناسب را در نظر بگیرید. ولی بهتر است که فعلا از این هم صرف نظر کنید و فرض کنید که ورودی حتما عدد صحیح یا اعشاری است، تا کار راحت تر شود. بعدا اگر خواستید، می توانید آنها را هم به کدهایتان اضافه کنید.

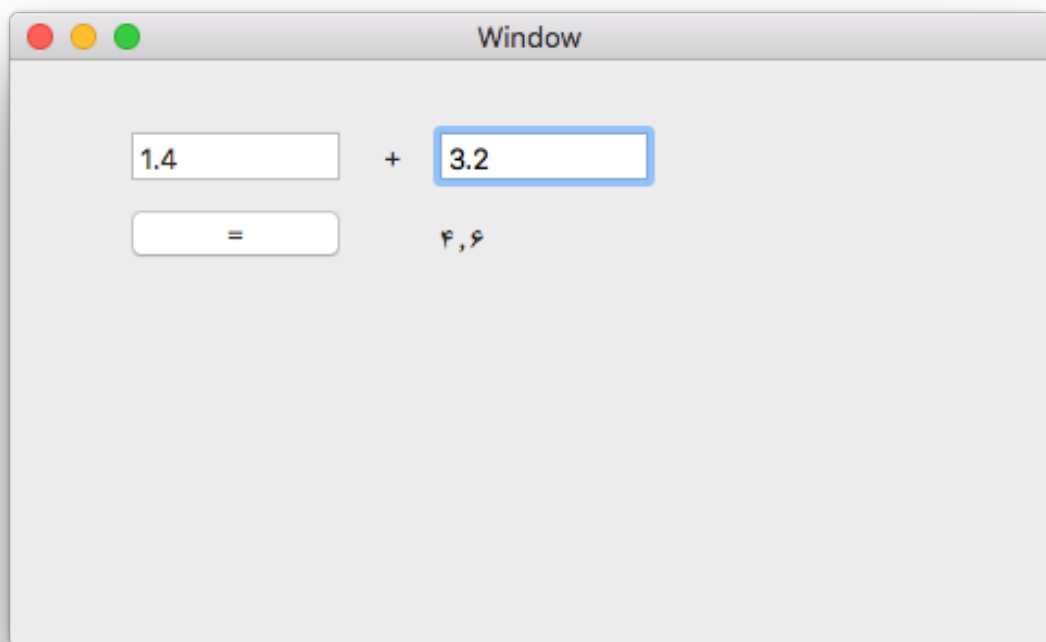
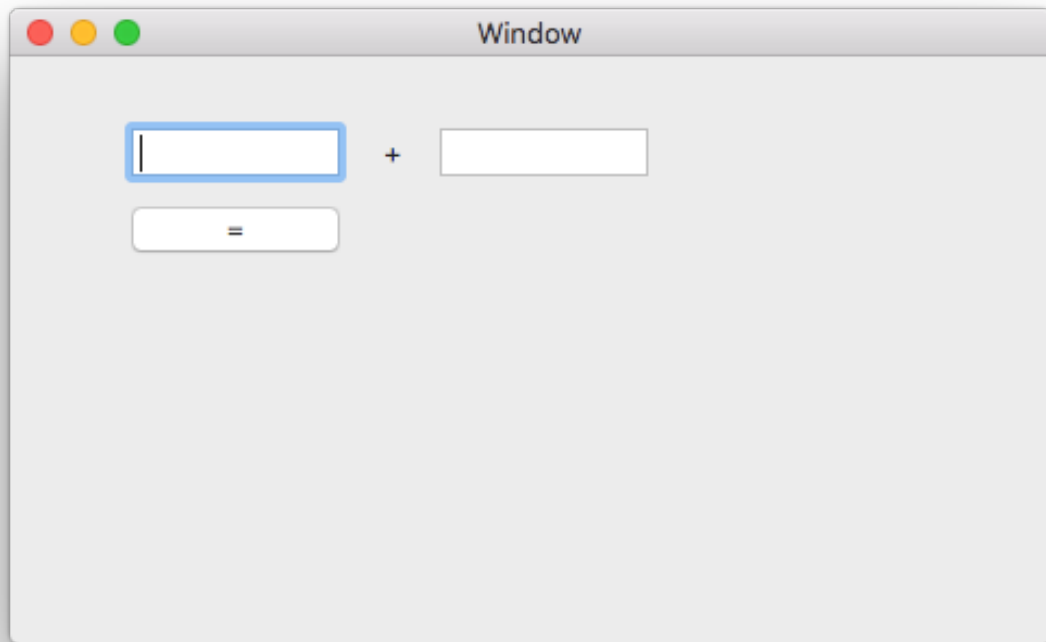


حالا باید ببینید چطور می‌شود از یک Text Field، یک عدد double بیرون کشید و چطور می‌توان جواب را درون Label گذاشت. خوشبختانه هردوی اینها از نوع NSTextField هستند که جزئی به نام doubleValue دارد که اگر بتواند از متن نوشته شده، چیزی به فرم عدد اعشاری در بیاورد، آن را در خودش نگه می‌دارد. بنابراین کافی است کدی شبیه این بنویسید:

`answer.doubleValue = firstNumber.doubleValue + secondNumber.doubleValue`



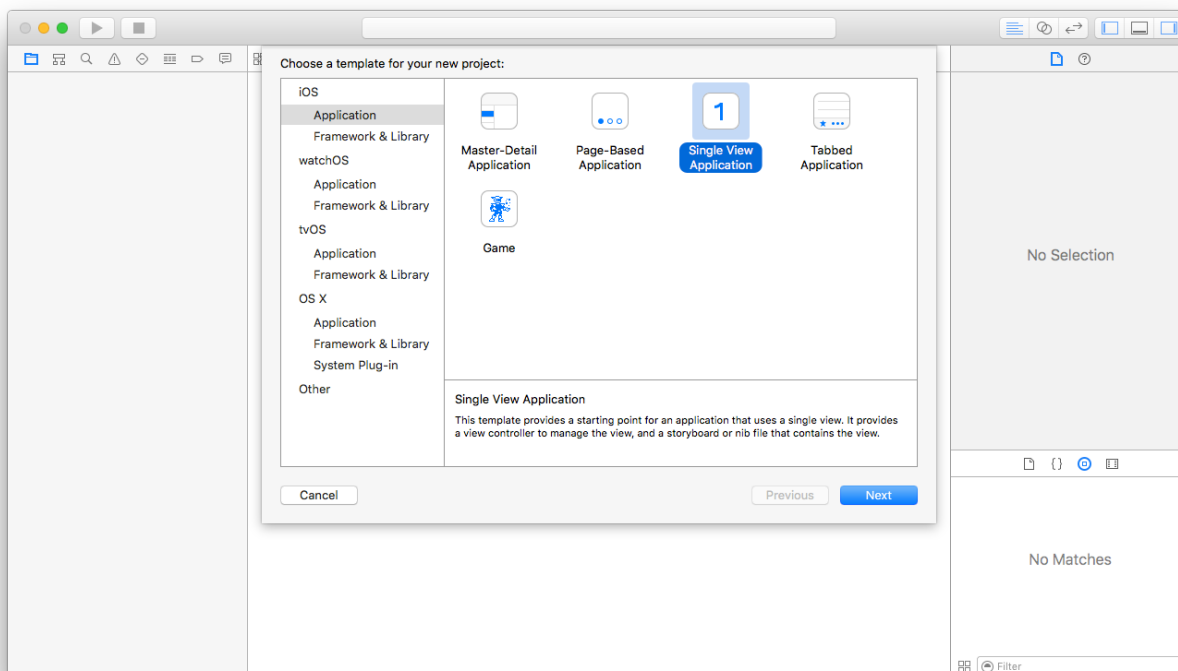
حالا برنامه شما کامل شده و می‌توانید آن را آزمایش کنید:



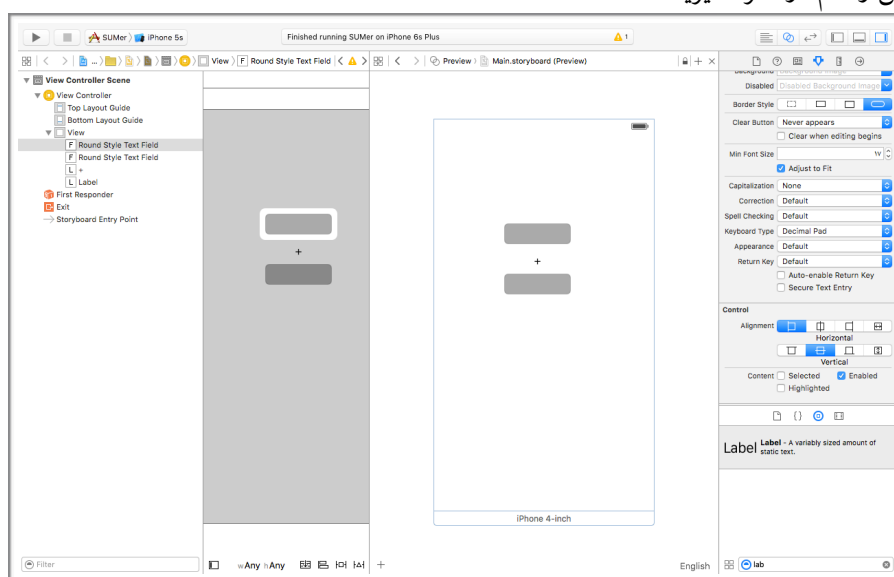
برنامه برای iOS

از آنجایی که برنامه ساختن برای OS X و iOS خیلی به هم شبیه است، در این قسمت همان برنامه بالا را برای iOS می سازیم و اینبار کمی بیشتر به جزئیات دقت می کنیم.

طبیعتا اول باید یک پروژه جدید بسازید. از آنجا که این برنامه قرار است فقط یک صفحه داشته باشد، اینبار نوع برنامه را Single View Application برای iOS بگذارید:

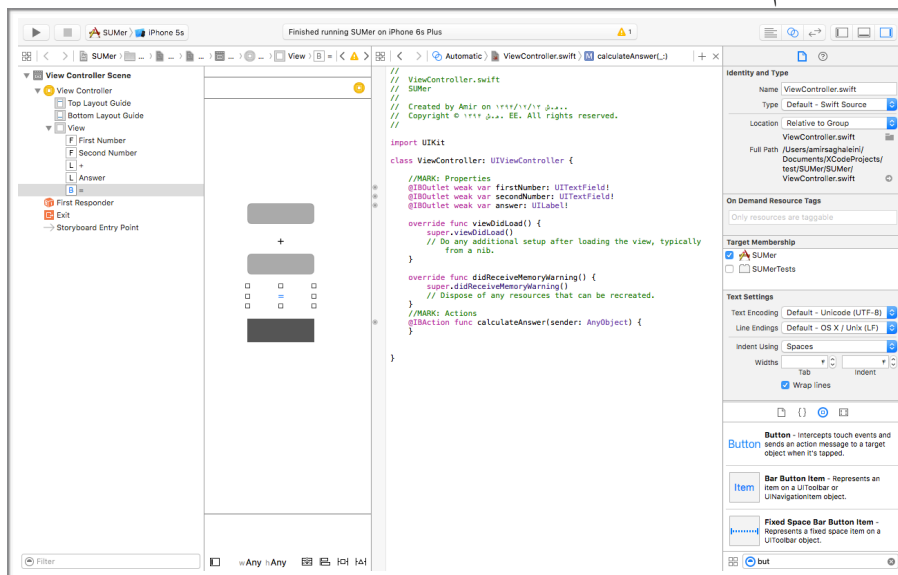


بقیه مراحل را شبیه قبل طی کنید تا جایی که واسط گرافیکی ساخته شود. هرچند دستگاه های iOS قابلیت اتصال کیبورد فیزیکی دارند و حتی آپد پرو، داخل جلد خودش یک کیبورد دارد، ولی فعلا فرض کنید که تنها راه وارد کردن اطلاعات، همان کیبورد نرم افزاری است. با این فرض، بد نیست که برای Text Field ها، نوع کیبورد را در پنل سمت راست Decimal Pad بگذارید تا فقط امکان وارد کردن عدد و ممیز وجود داشته باشد. البته اینجا هم احتمال دارد که کسی شیطنت کند و بیشتر از یک ممیز بگذارد. ولی فعلا این را هم در نظر نگیرید:



تفاوت دیگر تصویر بالا با مرحله مشابه در برنامه مک این است که برای بهتر دیده شدن کادرها در تصویر، رنگ آنها تغییر کرده است. اگر خواستید، می توانید این قبیل تغییرات را هم به سلیقه خودتان و با استفاده از پنل سمت راست اجرا کنید. علاوه بر آن، از

آنجا که در iOS، سایز صفحه متناسب با مدل دستگاه تغییر می‌کند، بهتر است موقع چیدن اجزا، ادیتور کمکی را برای نمایش دادن Preview از خروجی دستگاه مورد نظرتان تنظیم کنید تا بتوانید نتیجه را ببینید. البته Xcode برای این شرایط هم ابزارهایی دارد که به کمک آنها بشود یک طراحی را با تقریب خیلی خوبی برای همه سایزها تنظیم کرد، ولی فعلا با آنها کاری نداشته باشید. کار را ادامه بدهید تا اتصال‌ها هم برقرار شود و نتیجه‌ای شبیه تصویر زیر داشته باشید:



حالا کم‌کم بخش متفاوت ماجرا شروع می‌شود. اما چون تا همه چیز تکراری بوده، بد نیست برای تنوع، خالی نبودن کادرها را هم چک بکنید.

قبل از هرچیز، باید محتوای کادرها خوانده شود. شاید متوجه شده باشید که در iOS نوع کادرها بجای NSString، از جنس UITextField است و متاسفانه یا خوشبختانه doubleValue، intValue، stringValue و امثال اینها را ندارد، بلکه بجای آنها فقط یک text دارد که از نوع string است.

بنابراین باید یک شرط بگذارید که فقط اگر هردو خانه پر بودند، برود سراغ حساب و کتاب. احتمالا می‌دانید که مقدار خالی را با nil نشان می‌دهند، همچنین برای نا مساوی از != و برای «و»ی شرطی از && استفاده می‌شود. پس دستور شرط را که باید بعد از فشرده شدن کلید مساوی بررسی شود، باید چیزی شبیه به این بنویسید:

```
if firstNumber.text != nil && secondNumber.text != nil
```

حالا بعد از آن، دو آکولاد باز و بسته بگذارید تا بین آنها، دستور جمع نوشته شود.

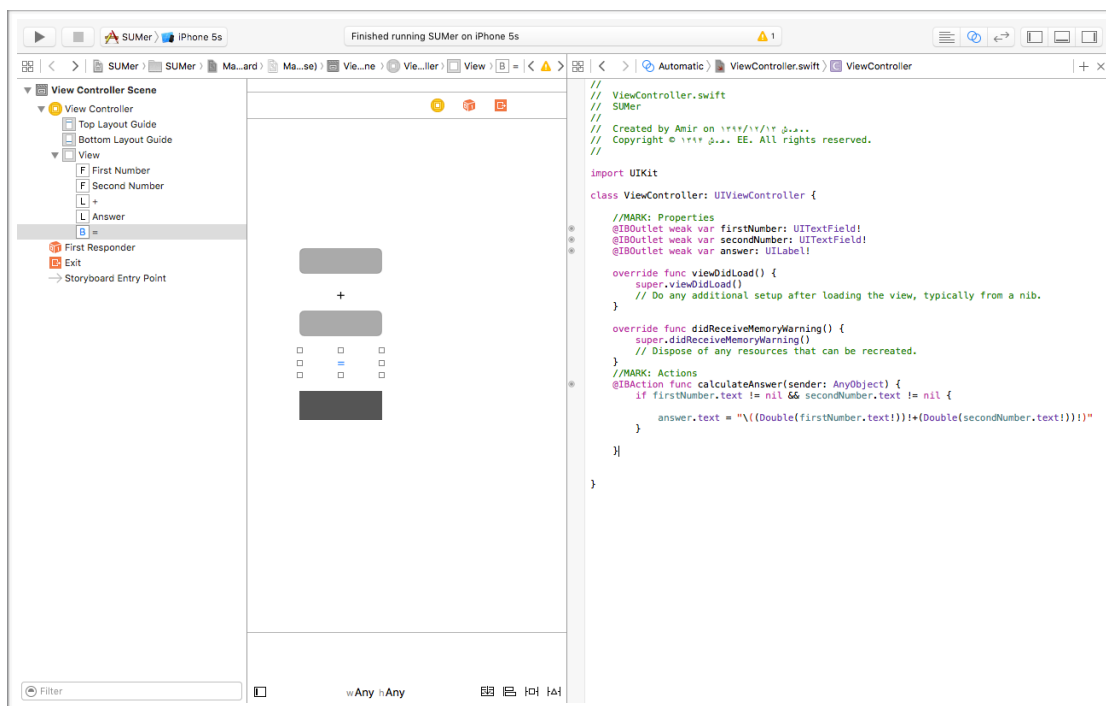
از اینجا داستان کمی سخت می‌شود! چون قرار است دو عدد با هم جمع شوند ولی هر سه خانه‌ای که قرار است ورودی و خروجی این جمع باشند، از جنس رشته متنی هستند و Swift هم خیلی به این چیزها حساس است. اما خوشبختانه راهی برای تبدیل این جنس‌ها به هم وجود دارد. ابتدا باید از تابع Double() استفاده کنید و متغیر متنی خودتان را بین پرانتزهایش بگذارید تا آن را به عدد تبدیل کند. اما کار به این سادگی نیست، چون Xcode به شما گیر خواهد داد که شاید چیزی که به این تابع می‌دهید خالی باشد. البته شما می‌دانید که قبلش این موضوع را چک کرده‌اید و خیالتان از این بابت راحت است، پس برای اینکه خیال Xcode هم راحت شود، این موضوع را با گذاشتن یک علامت تعجب در انتهای نام متغیر به Xcode اطلاع بدهید. و البته چون دو عددی که به دست می‌آید قرار است با هم جمع شود، لازم است بعد از پرانتز تابع Double هم یک علامت تعجب بگذارید تا به Xcode اطلاع دهید که مطمئن هستید که ورودی این تابع قابل تبدیل به نوع double است و خروجی آن هم nil نخواهد بود. پس تا اینجا جمع دو عدد به اینصورت نوشته می‌شود:

```
Double(firstNumber.text!)! + Double(secondNumber.text!)!
```

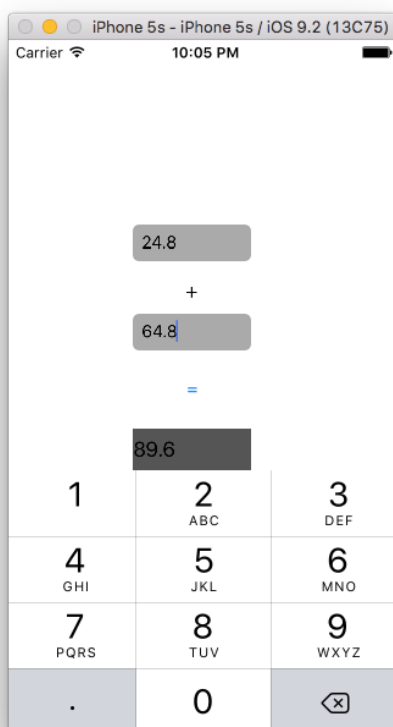
حالا باید این را به string تبدیل کنید و در answer بگذارید، برای اینکار از بلوک "()" استفاده می‌شود و عددی که باید به متن تبدیل شود، داخل پرانتزها قرار می‌گیرد. یعنی کد نهایی به این صورت خواهد بود:

```
answer.text = "\( Double(firstNumber.text!)! + Double(secondNumber.text!)! )"
```

و در نهایت کد شما هم باید چیزی شبیه به این بشود:



و اگر برنامه را با استفاده از کلید شبیه‌سازی که گوشه بالا سمت چپ قرار دارد، در آیفون یا آیپد دلخواهتان اجرا کنید، چیزی شبیه این خواهید دید:



البته اگر دوست داشته باشید، می‌توانید با استفاده از دستور `else if` یا `else`، برای شرایطی که یکی از خانه‌ها یا هردوی آنها خالی است هم کدی بنویسید که پیام خطای مناسب بدهد.

راستی احتمالاً متوجه شده‌اید که در حین کار، Xcode مرتباً کدها و دستوراتی که احتمالاً لازم دارید را پیشنهاد می‌دهد یا حتی خودش می‌نویسد و هرجایی هم که اشتباهی ببیند، آن را تذکر داده و سعی می‌کند راه درست کردن آن را پیشنهاد بدهد.

و حرف پایانی اینکه اگر آیفون یا آیپد دم دستتان نبود و دوست داشتی کمی با آن بازی کنید، وقتی شبیه‌ساز فعال است، می‌توانید از برنامه خودتان خارج شده و وارد صفحه Home آیفون شوید. تا یادم نرفته، بجز دکمه Home، سری دکمه و رفتار فیزیکی مثل

چرخاندن و تکان دادن هم برای آن در نظر گرفته‌اند از منوها قابل دسترسی هستند و موقع بازی کردن یا تست برنامه به دردتان می‌خورند.