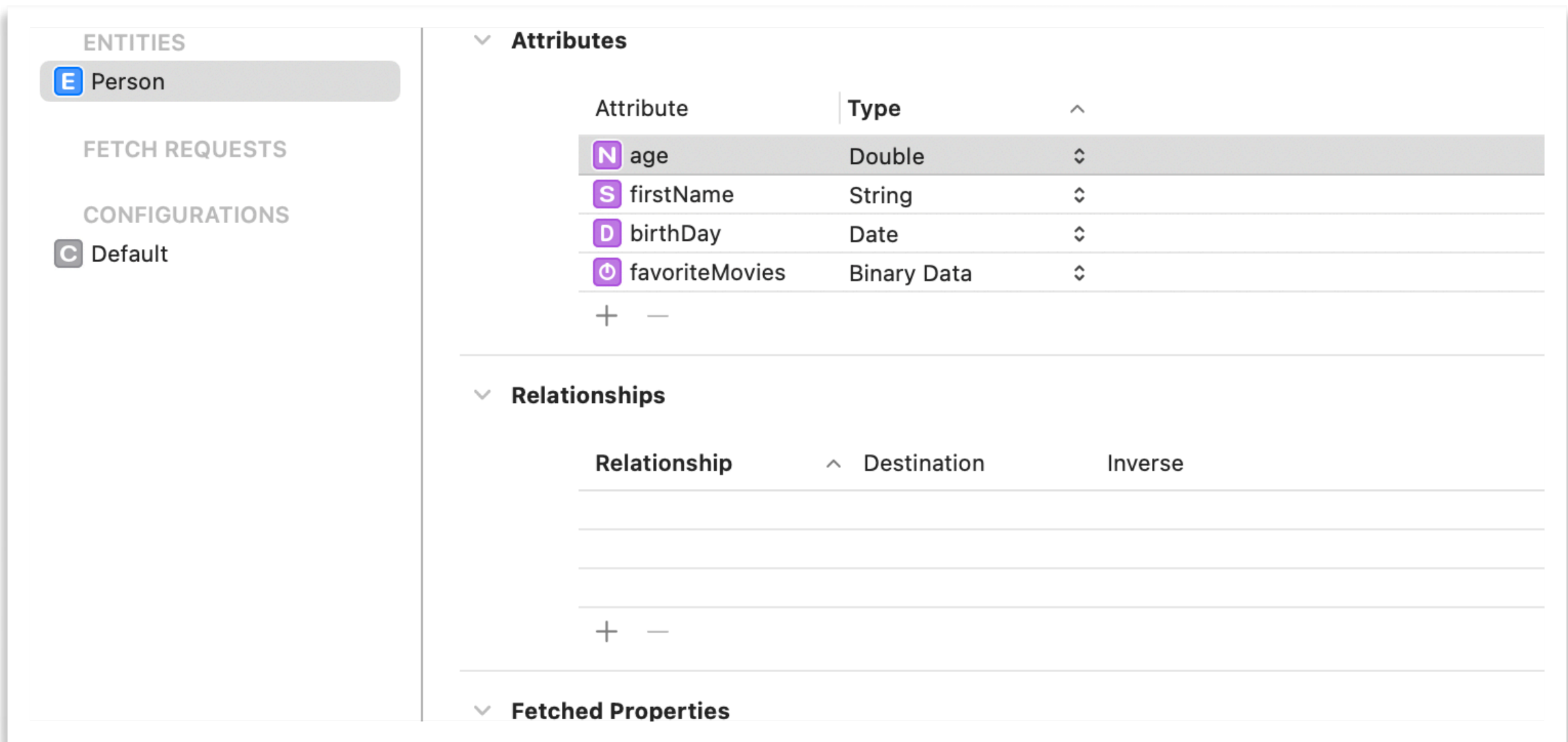


Core Data

אחסון נתוני אפליקציה
חלק ב'



נניח שנרצה לשמור באובייקט המנוהל שלנו Person, כמאפיין, מערך של "Movie" בשם favoriteMovies, איך נעשה זאת?

לשם כך אנחנו נתייחס לתכונה שלנו כ מידע בינארי, ונשתמש ב JsonDecoder/JsonEncoder כדי להמיר אובייקטים של סוויפט למידע בינארי, והפוך -> מידע בינארי לאובייקטים

ראשית נוסיף את התכונה לאובייקט המנוהל עם הסוג **Binary Data** שנית נפתח struct חדש Movie כאשר נירש מ **Codable** (זאת מכיוון שנרצה בשלבים הבאים להמיר את האובייקט למידע בינארי והפוך)

נפתח עוד struct בשם FavoriteMovies שגם הוא יורש מ Codable ומחזיק מערך של Movie כתכונה

Movie.class

```
7
8 import Foundation
9
10 struct Movie : Codable {
11     let title:String
12     let genre:String
13 }
14
```

FavoriteMovies.class

```
7
8 import Foundation
9
10 struct FavoriteMovies : Codable {
11     let movies:[Movie]
12 }
13
```

```
12 @objc(Person)
13 public class Person: NSObject {
14
15
16     func getFavoriteMovies() -> FavoriteMovies? {
17         guard let data = self.favoriteMovies else {return nil}
18         let decoder = JSONDecoder()
19         guard let favmovies = try? decoder.decode(FavoriteMovies.self, from: data) else {return nil}
20         return favmovies
21     }
22
23     func addFavoriteMovie(movie:Movie) {
24         var newFavorites = getFavoriteMovies()
25
26         // if user has no favorites, create a new favoriteMovies object
27         // with the movie to be added inside
28
29         // else, append the movie to the existing favorites movies array
30         if newFavorites == nil {
31             newFavorites = FavoriteMovies(movies: [movie])
32         }else {
33             newFavorites!.movies.append(movie)
34         }
35         // finally, encode the favorites object to Binary data
36         // and assign the result to self.favoriteMovies
37         let encoder = JSONEncoder()
38         guard let data = try? encoder.encode(newFavorites) else {return}
39         self.favoriteMovies = data
40     }
41 }
42
```

עכשיו נטפל בחלק של הוספת סרט חדש ל favoriteMovies ושליפת כל הfavoriteMovies של Person מסוים.

מאחר ואלו שיטות שמתעסקות עם אובייקט מנוהל Person ספציפי, נכתוב אותם במחלקה של Person בקובץ **Person + CoreDataClass**

getFavoriteMovies

קודם נכתוב getter שעושה המרה ממידע בינארי לאובייקט FavoriteMovies, השיטה תחזיר אובייקט אופציונלי מסוג זה.

בתוך השיטה נבדוק האם לPerson שלנו קיימים בכלל סרטים Favorites אם לא, אז המידע הבינארי יהיה nil ולכן לא נתקדם הלאה, במקרה ויש, נשתמש ב JsonDecoder כדי להמיר את המידע הבינארי לאובייקט סוויפט FavoriteMovies, ונחזיר את התוצאה.

addFavoriteMovie

עכשיו נוסיף שיטה נוספת למחלקה הזו, כדי להוסיף סרט חדש לFavorites של אותו Person השיטה תקבל Movie בפרמטרים.

נשתמש ב getter שכתבנו כדי לשלוף את הסרטים הקיימים במועדפים של אותו Person, ונבדוק, אם קיבלנו חזרה nil, סימן שאין לו עדיין מועדפים ובמקרה זה ניצור מופע חדש של FavoriteMovies כאשר נעביר לו בבנאי מערך עם אובייקט יחיד - Movie אותו נרצה להוסיף.

אם הוא לא nil, סימן שיש לPerson מועדפים, לכן במקרה זה נוסיף למערך הסרטים הקיים את הסרט.

לבסוף, בכל מקרה, נרצה להמיר חזרה את האובייקט שלנו למידע בינארי כדי שנוכל להקצות אותו לתכונת המחלקה favoriteMovies, לשם כך נשתמש ב JsonEncoder ואת התוצאה נקצה למחלקה.

לבסוף, נוסיף שיטה **addMovieToFavorites**

ב**CoreDataManager** המקבלת שם פרטי של Person וסרט Movie מחפשת את הPerson הראשון בנתוני האפליקציה שתואם לשם הזה, ומוסיפה את הסרט עם השיטה משקופית קודמת לאותו Person

```
func addMovieToFavorites(firstName:String,movie:Movie) {
    guard let person = (fetchPersons().first{$0.firstName == firstName}) else {
        print("no person found with name \(firstName)")
        return
    }
    person.addFavoriteMovie(movie: movie)
    saveContext()
}
```

חיפוש Person הראשון עם השם הפרטי "Nadav" והדפסת כל הסרטים המועדפים של אותו Person

שימוש בשיטה, נוסיף לPerson עם השם הפרטי "Nadav" סרט חדש למועדפים!

```
override func viewDidLoad() {
    super.viewDidLoad()
    let manager = CoreDataManager.shared

    guard let person = (manager.fetchPersons().first{$0.firstName == "Nadav"}) else {return}
    print(person.getFavoriteMovies())

}
```

```
override func viewDidLoad() {
    super.viewDidLoad()
    let movie = Movie(title: "Superman", genre: "Action")
    let manager = CoreDataManager.shared
    manager.addMovieToFavorites(firstName: "Nadav", movie: movie)
```

```
}
```