



Programming Fundamentals

Inheritance: Improving Structure

Produced Dr. Siobhán Drohan
by: Ms. Mairéad Meagher
Ms Siobhan Roche

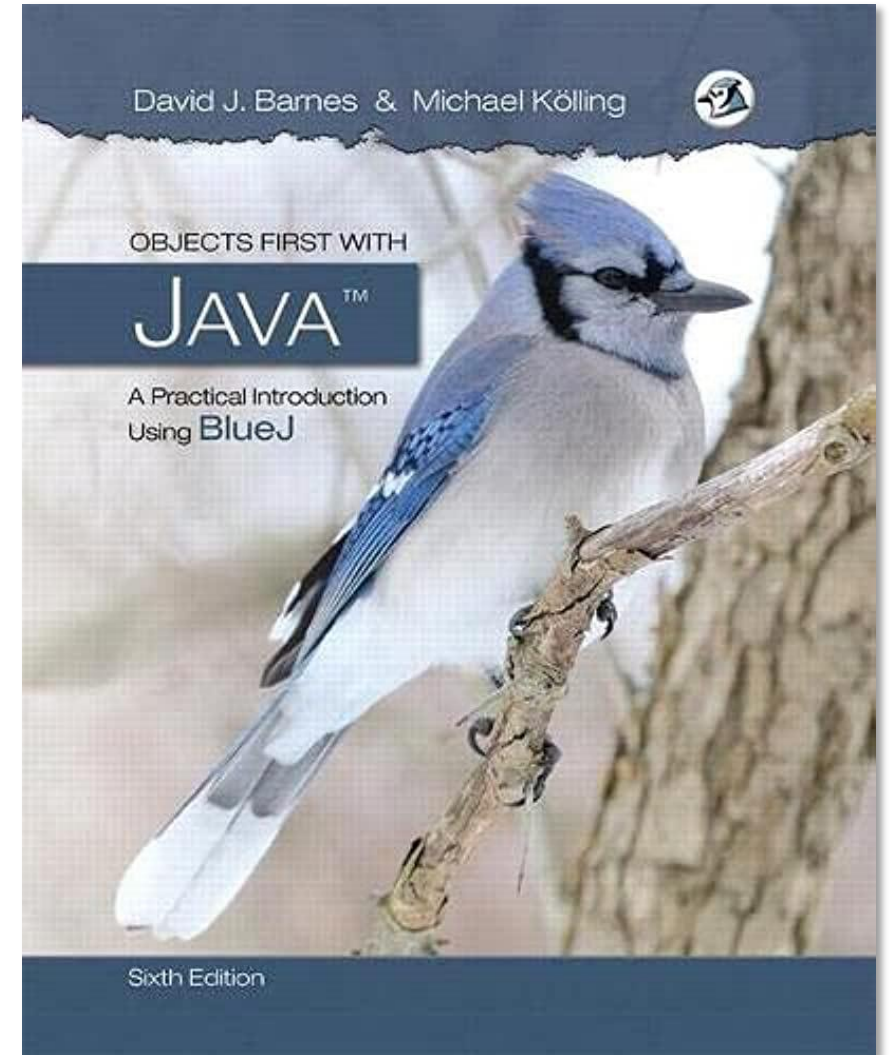
Agenda

- Social Network v1



Lectures & Labs

- This weeks lectures and labs are based on examples in:
- **Objects First with Java**
 - A Practical Introduction using BlueJ, © *David J. Barnes, Michael Kölling*



Social Network V3.0 – Recap

- A small, prototype SOCIAL NETWORK.
- Supports a **News Feed** with posts.
- POSTS:
 - **MessagePost:**
 - multi-line text message.
 - Operations
 - e.g., add, display and remove.



Social Network V3.0 – Recap

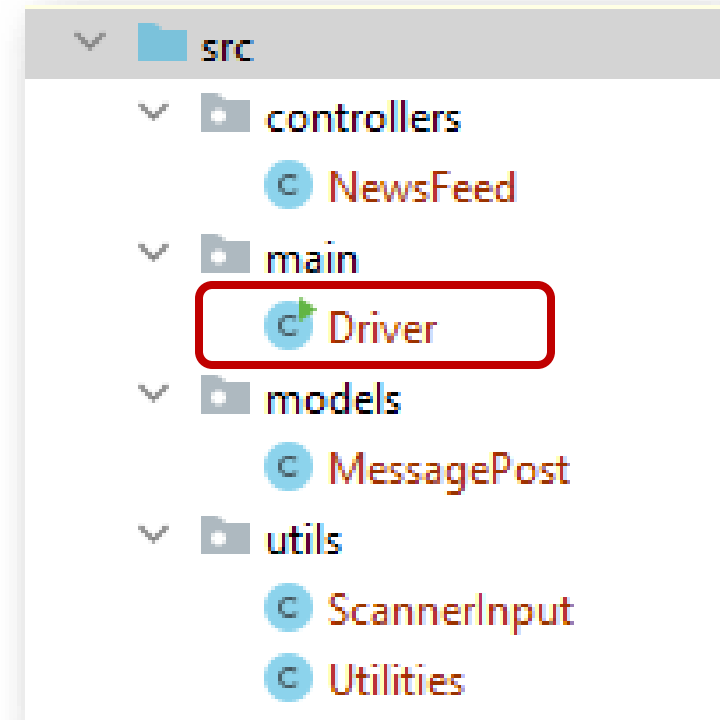
Social Network Menu

-
- 1) Add a Message Post
- 2) List all Posts
- 3) Update a Message Post
- 4) Delete a Message Post
-

- 5) Save Posts
- 6) Load Posts
-

0) Exit

==>>



Social Network V3.0 – Recap

```
C MessagePost
m MessagePost(String, String)
m getAuthor(): String
m setAuthor(String): void
m getMessage(): String
m setMessage(String): void
m getLikes(): int
m setLikes(int): void
m display(): String
f author: String = ""
f message: String = ""
f likes: int = 0
```

```
src
├── controllers
│   └── NewsFeed
├── main
│   └── Driver
├── models
│   └── MessagePost
└── utils
    ├── ScannerInput
    └── Utilities
```

```
Social Network Menu
-----
1) Add a Message Post
2) List all Posts
3) Update a Message Post
4) Delete a Message Post
-----
5) Save Posts
6) Load Posts
-----
0) Exit
==>>
```

Social Network V3.0 – Recap

Social Network Menu

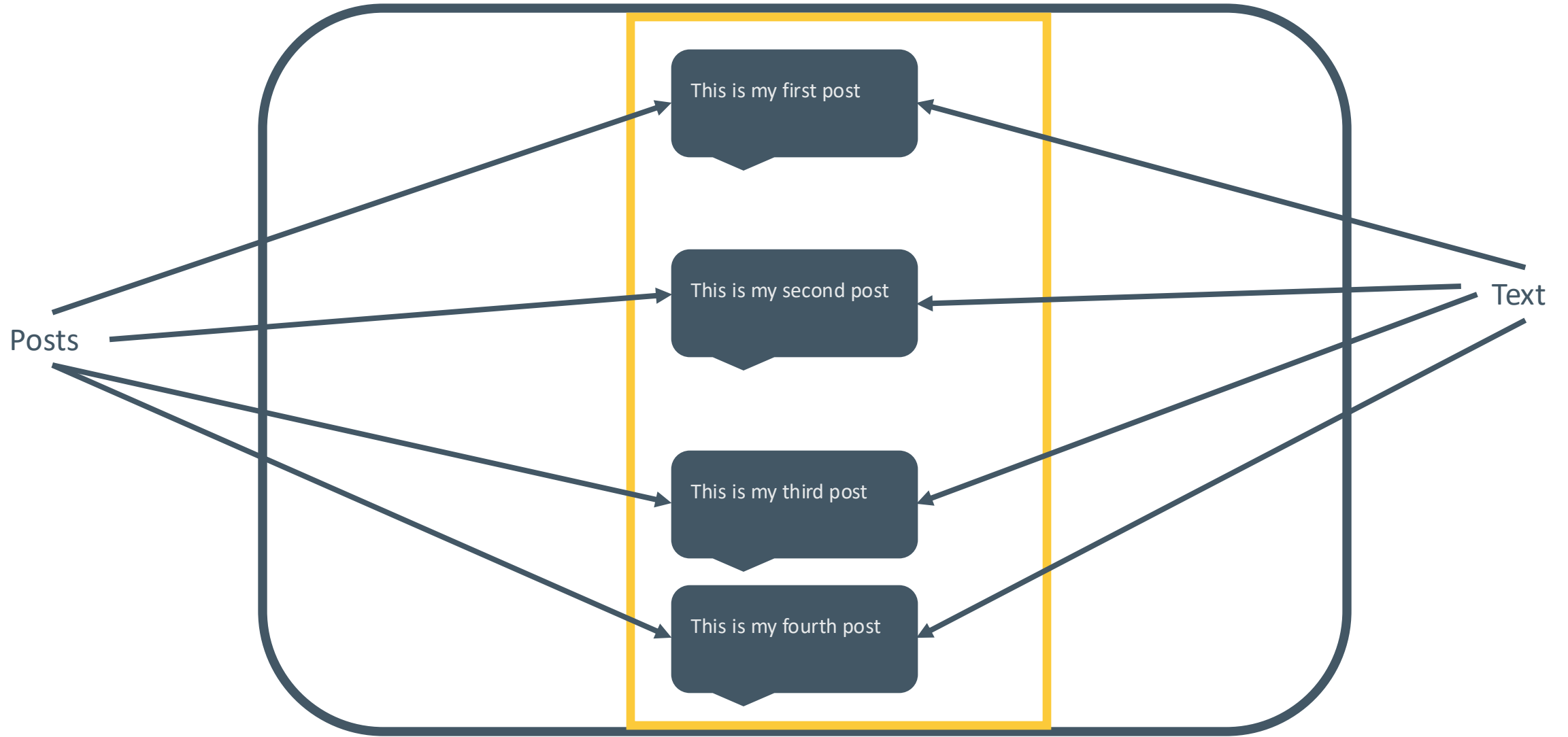
-
- 1) Add a Message Post
- 2) List all Posts
- 3) Update a Message Post
- 4) Delete a Message Post
-
- 5) Save Posts
- 6) Load Posts
-
- 0) Exit

==>>

```
C NewsFeed
  m NewsFeed()
  m addPost(MessagePost): boolean
  m show(): String
  m deleteMessagePost(int): MessagePost
  m updateMessagePost(int, String, String): boolean
  m findMessagePost(int): MessagePost
  m numberOfPosts(): int
  m load(): void
  m save(): void
  m isValidIndex(int): boolean
  f posts: ArrayList<MessagePost>
```

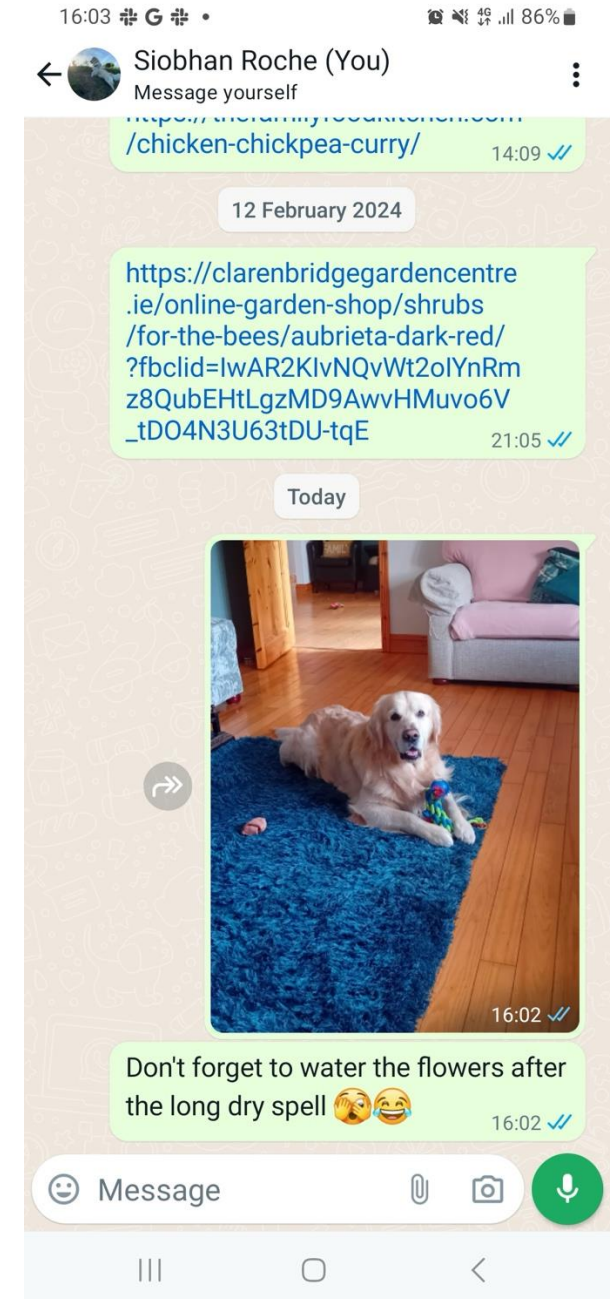
```
src
├── controllers
│   └── NewsFeed
├── main
│   └── Driver
├── models
│   └── MessagePost
├── utils
│   ├── ScannerInput
│   └── Utilities
```

Social Network

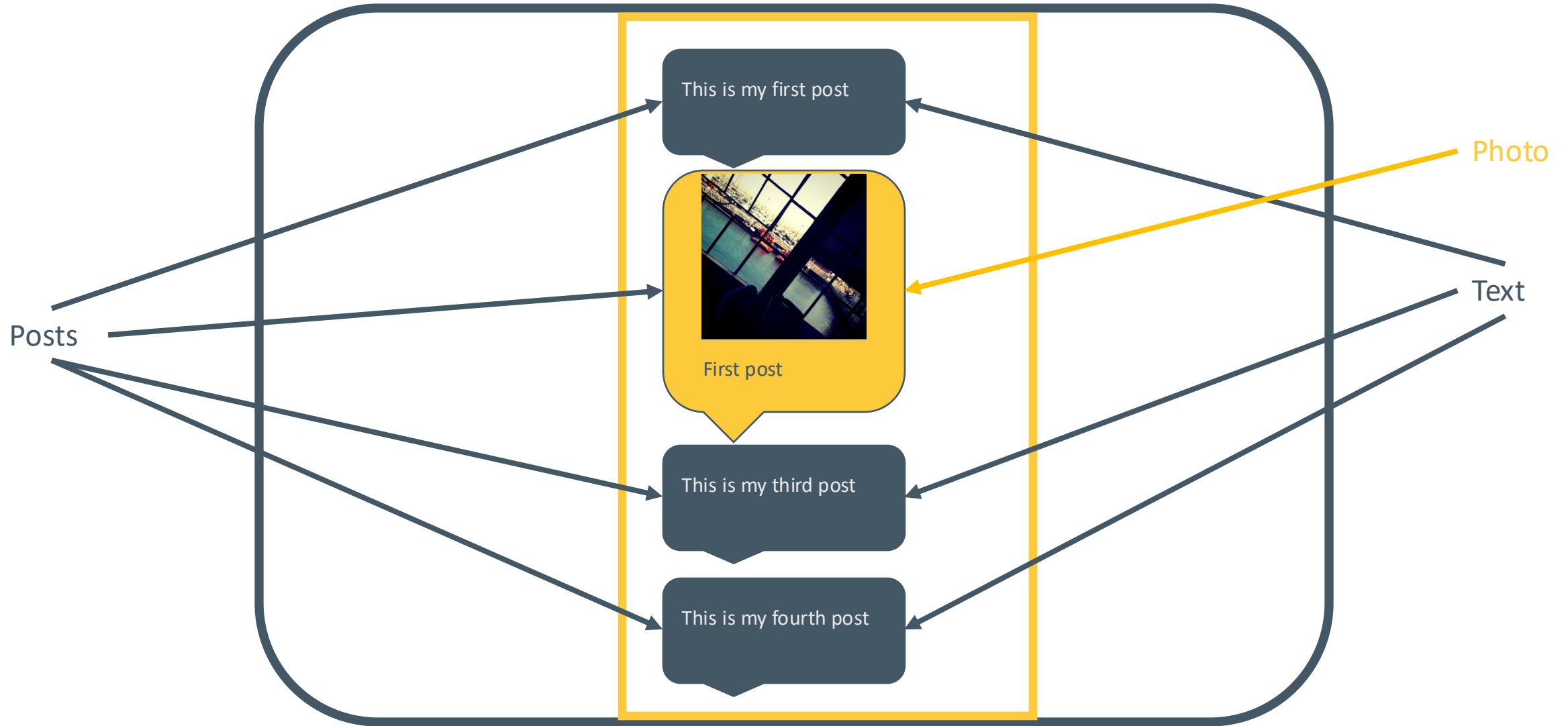


Social Network v4.0 – lets move it on!

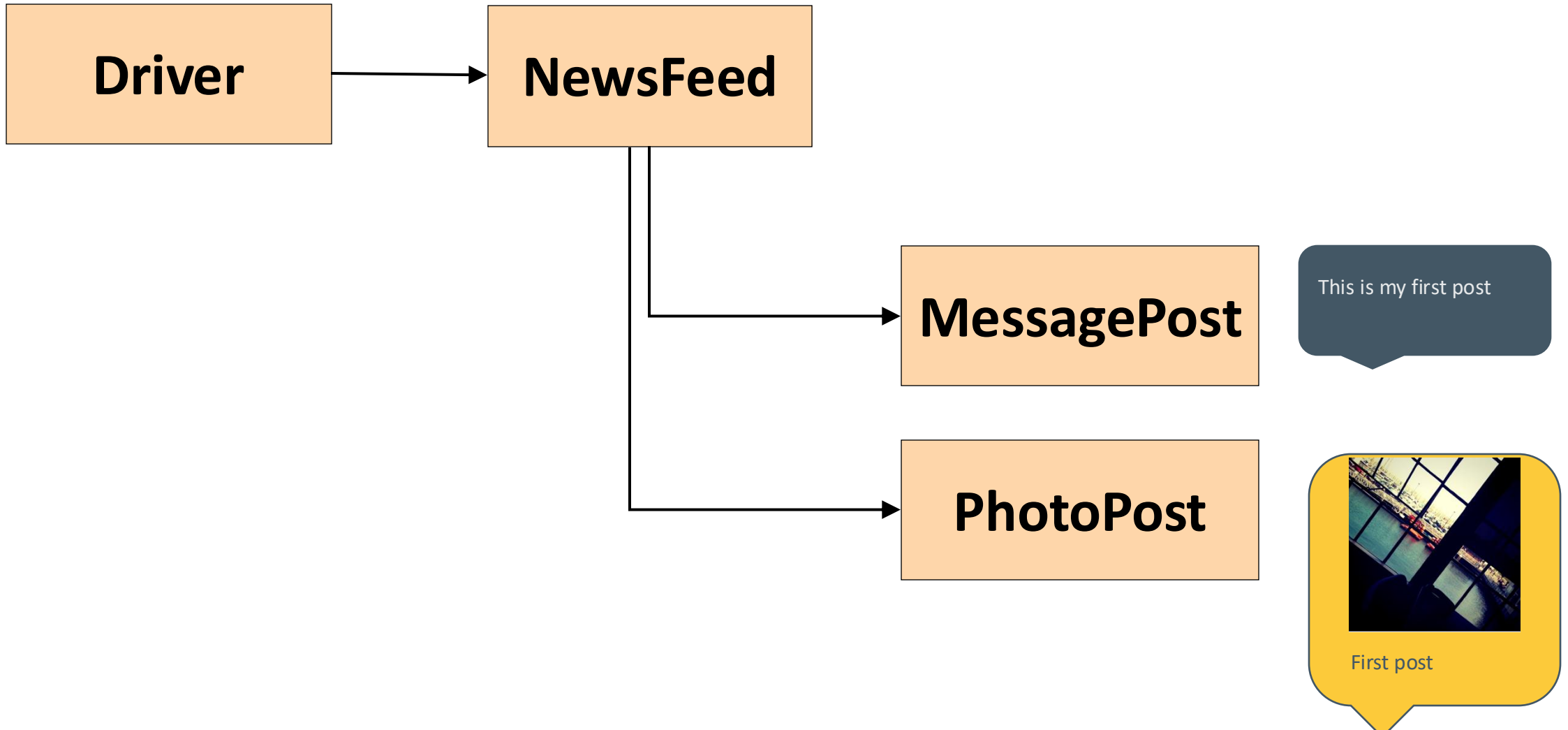
- A small, prototype SOCIAL NETWORK.
- Supports a News Feed with posts.
- **POSTS:**
 - **MessagePost:**
 - multi-line text messages.
 - **PhotoPost:**
 - photo and caption.
- **Operations**
 - e.g., search, display and remove.



Social Network



Social Network V4.0 – Class Diagram



Social Network V4.0 – Two Object Types

MessagePost

author	""
message	""
likes	0

This is my first post

MessagePost:
multi-line text
message.

PhotoPost

author	""
caption	""
filename	""
likes	0

PhotoPost:
photo and caption.



First post

Social Network V4.0 – Class Structure

MessagePost

- m MessagePost(String, String)
- m getAuthor(): String
- m setAuthor(String): void
- m getMessage(): String
- m setMessage(String): void
- m getLikes(): int
- m setLikes(int): void
- m display(): String
- f author: String = ""
- f message: String = ""
- f likes: int = 0



PhotoPost

- m PhotoPost(String, String, String)
- m getAuthor(): String
- m setAuthor(String): void
- m getCaption(): String
- m setCaption(String): void
- m setFilename(String): void
- m getFilename(): String
- m getLikes(): int
- m setLikes(int): void
- m display(): String
- f author: String = ""
- f caption: String = ""
- f filename: String = ""
- f likes: int = 0



Social Network V4.0 – Class Structure

```
C MessagePost
m MessagePost(String, String)
m getAuthor(): String
m setAuthor(String): void
m getMessage(): String
m setMessage(String): void
m getLikes(): int
m setLikes(int): void
m display(): String
f author: String = ""
f message: String = ""
f likes: int = 0
```



```
C PhotoPost
m PhotoPost(String, String, String)
m getAuthor(): String
m setAuthor(String): void
m getCaption(): String
m setCaption(String): void
m setFilename(String): void
m getFilename(): String
m getLikes(): int
m setLikes(int): void
m display(): String
f author: String = ""
f caption: String = ""
f filename: String = ""
f likes: int = 0
```



author:
code duplication

Social Network V4.0 – Class Structure

```
C MessagePost
m MessagePost(String, String)
m getAuthor(): String
m setAuthor(String): void
m getMessage(): String
m setMessage(String): void
m getLikes(): int
m setLikes(int): void
m display(): String
f author: String = ""
f message: String = ""
f likes: int = 0
```

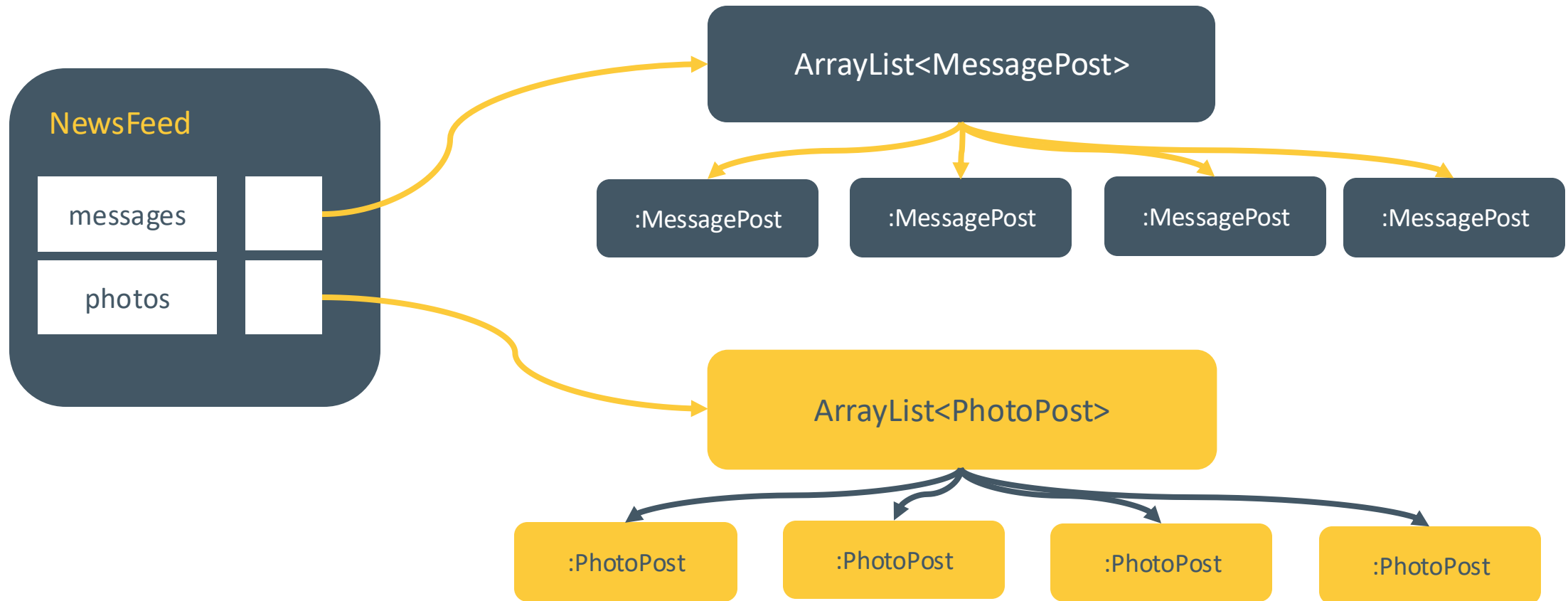


```
C PhotoPost
m PhotoPost(String, String, String)
m getAuthor(): String
m setAuthor(String): void
m getCaption(): String
m setCaption(String): void
m setFilename(String): void
m getFilename(): String
m getLikes(): int
m setLikes(int): void
m display(): String
f author: String = ""
f caption: String = ""
f filename: String = ""
f likes: int = 0
```



likes:
code duplication

Social Network V4.0 – CRUD on TWO ArrayLists



Social Network V3.0 versus V4.0 (NewsFeed)

V3.0

```
c NewsFeed
  m NewsFeed()
  m addPost(MessagePost): boolean
  m show(): String
  m deleteMessagePost(int): MessagePost
  m updateMessagePost(int, String, String): boolean
  m findMessagePost(int): MessagePost
  m numberOfPosts(): int
  m load(): void
  m save(): void
  m isValidIndex(int): boolean
  f posts: ArrayList<MessagePost>
```

V4.0

```
c NewsFeed
  m NewsFeed()
  m addMessagePost(MessagePost): boolean
  m addPhotoPost(PhotoPost): boolean
  m show(): String
  m showPhotoPosts(): String
  m showMessagePosts(): String
  m deleteMessagePost(int): MessagePost
  m deletePhotoPost(int): PhotoPost
  m updateMessagePost(int, String, String): boolean
  m updatePhotoPost(int, String, String, String): boolean
  m findMessagePost(int): MessagePost
  m findPhotoPost(int): PhotoPost
  m numberOfMessagePosts(): int
  m numberOfPhotoPosts(): int
  m load(): void
  m save(): void
  m isValidMessagePostIndex(int): boolean
  m isValidPhotoPostIndex(int): boolean
  f messagePosts: ArrayList<MessagePost>
  f photoPosts: ArrayList<PhotoPost>
```

Social Network V3.0 versus V4.0 (NewsFeed)

V3.0

```
NewsFeed
m NewsFeed()
m addPost(MessagePost): boolean
m show(): String
m deleteMessagePost(int): MessagePost
m updateMessagePost(int, String, String): boolean
m findMessagePost(int): MessagePost
m numberOfPosts(): int
m load(): void
m save(): void
m isValidIndex(int): boolean
f posts: ArrayList<MessagePost>
```

To handle the two ArrayLists, our NewsFeed methods have grown significantly e.g. delete now has two methods!

V4.0

```
NewsFeed
m NewsFeed()
m addMessagePost(MessagePost): boolean
m addPhotoPost(PhotoPost): boolean
m show(): String
m showPhotoPosts(): String
m showMessagePosts(): String
m deleteMessagePost(int): MessagePost
m deletePhotoPost(int): PhotoPost
m updateMessagePost(int, String, String): boolean
m updatePhotoPost(int, String, String, String): boolean
m findMessagePost(int): MessagePost
m findPhotoPost(int): PhotoPost
m numberOfMessagePosts(): int
m numberOfPhotoPosts(): int
m load(): void
m save(): void
m isValidMessagePostIndex(int): boolean
m isValidPhotoPostIndex(int): boolean
f messagePosts: ArrayList<MessagePost>
f photoPosts: ArrayList<PhotoPost>
```

Social Network V3.0 versus V4.0 (Driver)

V3.0

```
Driver
Driver()
main(String[]): void
mainMenu(): int
runMenu(): void
addMessagePost(): void
showPosts(): void
updateMessagePost(): void
deleteMessagePost(): void
savePosts(): void
loadPosts(): void
newsFeed: NewsFeed = new NewsFeed()
```

```
Social Network Menu
-----
1) Add a Message Post
2) List all Posts
3) Update a Message Post
4) Delete a Message Post
-----
5) Save Posts
6) Load Posts
-----
0) Exit
==>>
```

V4.0

```
Driver
Driver()
main(String[]): void
mainMenu(): int
runMenu(): void
addMessagePost(): void
updateMessagePost(): void
deleteMessagePost(): void
addPhotoPost(): void
updatePhotoPost(): void
deletePhotoPost(): void
showPosts(): void
showMessagePosts(): void
showPhotoPosts(): void
savePosts(): void
loadPosts(): void
newsFeed: NewsFeed = new NewsFeed()
```

```
Social Network Menu
-----
1) Add a Message Post
2) Update a Message Post
3) Delete a Message Post
-----
4) Add a Photo Post
5) Update a Photo Post
6) Delete a Photo Post
-----
7) List all Posts
-----
8) Save Posts
9) Load Posts
-----
0) Exit
==>>
```

Social Network V3.0 versus V4.0 (Driver)

V3.0

```
Driver
m Driver()
m main(String[]): void
m MainMenu(): int
m runMenu(): void
m addMessagePost(): void
m showPosts(): void
m updateMessagePost(): void
m deleteMessagePost(): void
m savePosts(): void
m loadPosts(): void
f newsFeed: NewsFeed = new NewsFeed()
```

```
Social Network Menu
-----
1) Add a Message Post
2) List all Posts
3) Update a Message Post
4) Delete a Message Post
-----
5) Save Posts
6) Load Posts
-----
0) Exit
==>>
```

To handle the two ArrayLists, we have added new Driver methods and new options on the menu

V4.0

```
Driver
m Driver()
m main(String[]): void
m MainMenu(): int
m runMenu(): void
m addMessagePost(): void
m updateMessagePost(): void
m deleteMessagePost(): void
m addPhotoPost(): void
m updatePhotoPost(): void
m deletePhotoPost(): void
m showPosts(): void
m showMessagePosts(): void
m showPhotoPosts(): void
m savePosts(): void
m loadPosts(): void
f newsFeed: NewsFeed = new NewsFeed()
```

```
Social Network Menu
-----
1) Add a Message Post
2) Update a Message Post
3) Delete a Message Post
-----
4) Add a Photo Post
5) Update a Photo Post
6) Delete a Photo Post
-----
7) List all Posts
-----
8) Save Posts
9) Load Posts
-----
0) Exit
==>>
```

Shop Network V4.0 – NewsFeed – show() method

```
public String show() {  
    String str = "";  
  
    for(MessagePost messagePost : messagePosts) {  
        str += messagePosts.indexOf(messagePost) + ": " + messagePost.display() + "\n";  
    }  
  
    for(PhotoPost photoPost : photoPosts) {  
        str += photoPosts.indexOf(photoPost) + ": " + photoPost.display() + "\n";  
    }  
  
    if (str.isEmpty()){  
        return "No Posts";  
    }  
    else {  
        return str;  
    }  
}
```

```
public class NewsFeed {  
  
    private ArrayList<MessagePost> messagePosts;  
    private ArrayList<PhotoPost> photoPosts;
```

Shop Network V4.0 – NewsFeed – show() method

```
public String show() {  
    String str = "";  
  
    for(MessagePost messagePost : messagePosts) {  
        str += messagePosts.indexOf(messagePost) + ": " + messagePost.display() + "\n";  
    }  
  
    for(PhotoPost photoPost : photoPosts) {  
        str += photoPosts.indexOf(photoPost) + ": " + photoPost.display() + "\n";  
    }  
  
    if (str.isEmpty()){  
        return "No Posts";  
    }  
    else {  
        return str;  
    }  
}
```

```
public class NewsFeed {  
  
    private ArrayList<MessagePost> messagePosts;  
    private ArrayList<PhotoPost> photoPosts;
```

Problem: The posts are not displayed in the order they were added, all the messages are displayed first, followed by the photos.

This is my first post

This is my 2nd post

This is my 3rd post



First Post



Second Post

Critique of Social Network V4.0

- Code duplication:
 - **MessagePost** and **PhotoPost** classes very similar (large parts are identical)
 - makes maintenance difficult/more work
 - introduces danger of bugs through incorrect maintenance
- Code duplication in **NewsFeed** and **Driver** classes as well.
- We need a better solution → **Inheritance!**

Now Let's Introduce Inheritance!