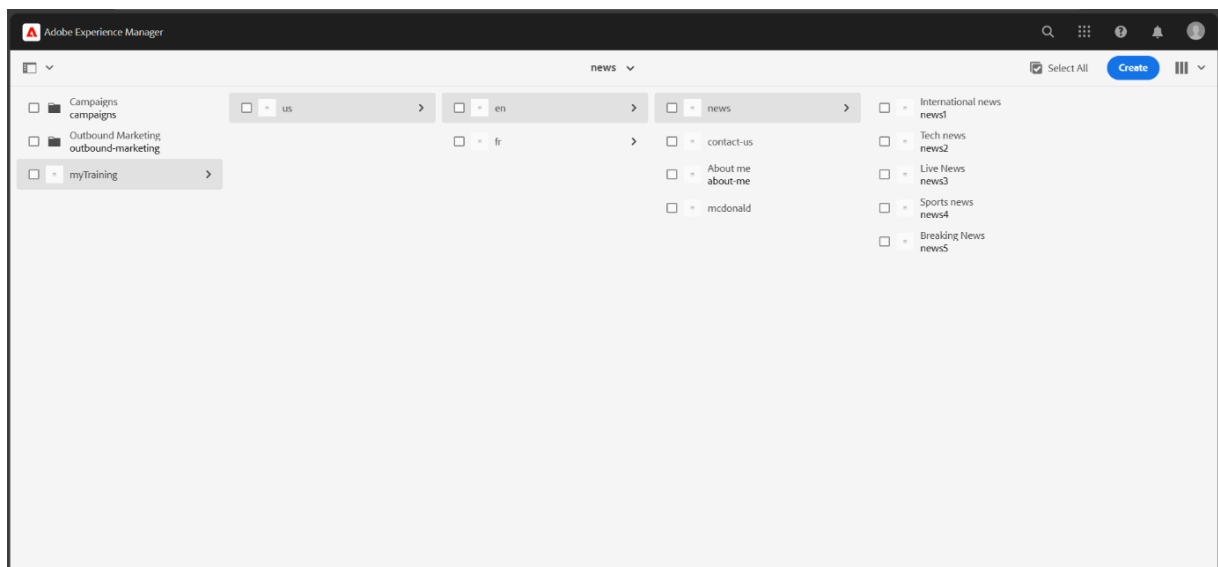


AEM – 5 News Article Pages and Printing in Log File

Step 1: Create 5 News Article Pages

1. Navigate to AEM Sites → /content/us/en/news.
2. Click Create → Page, choose the News Template, and enter details:
 - Title: Unique for each article (e.g., "Tech Innovations 2025", "Global Economy Update", "AI in Healthcare", etc.).
 - Page Name: news-1, news-2, etc.
3. Open each page and add the News Component:
 - Title: Set appropriate news title.
 - News Details: Add content.
 - Published Date: Set date.
4. Save and publish.

Screenshot



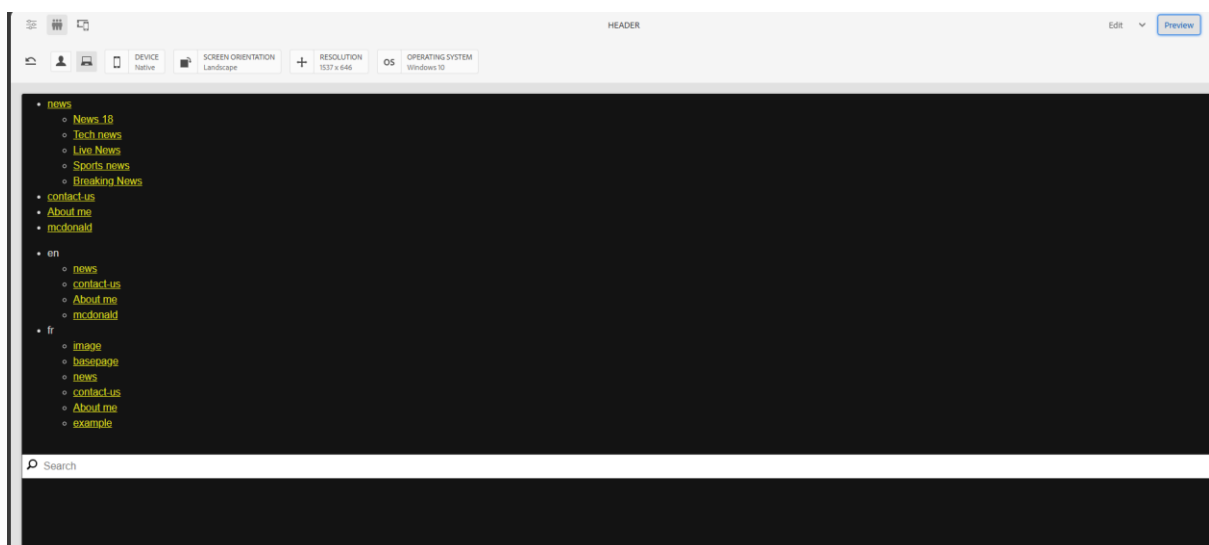
Step 2: Create Header Experience Fragment (XF)

1. Navigate to AEM → Experience Fragments.
2. Click Create → Experience Fragment.

3. Choose Header Template, set the name as News Header, and save it at /content/experience-fragments/us/en/news-header.
4. Open the News Header XF, add:
 - Navigation Component (links to "News", "About Me", and "Contact Us").
5. Save and publish.

Screenshot

- Experience Fragment



- Header Display



Step 3: Create "About Me" and "Contact Us" Pages

1. Navigate to /content/us/en/ → Create Page for:
 - About Me Page
 - Contact Us Page
2. Edit "About Me" Page:

- Image Component: Journalist photo.
- Text Component: Biography.
- Title Component: "About Me".

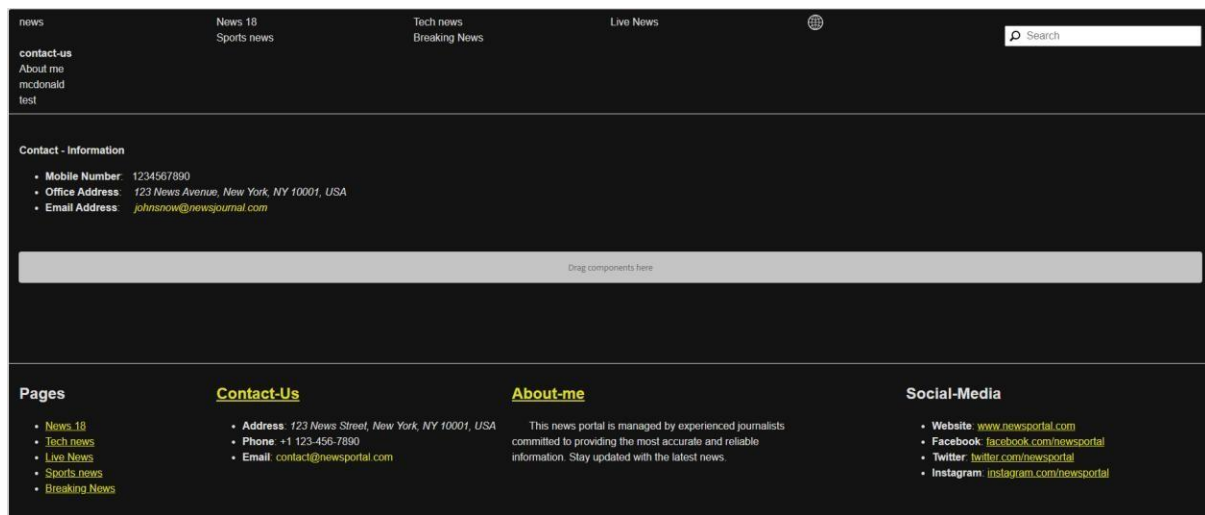
3. Edit "Contact Us" Page:

- Text Component: Phone number, email, and office address.

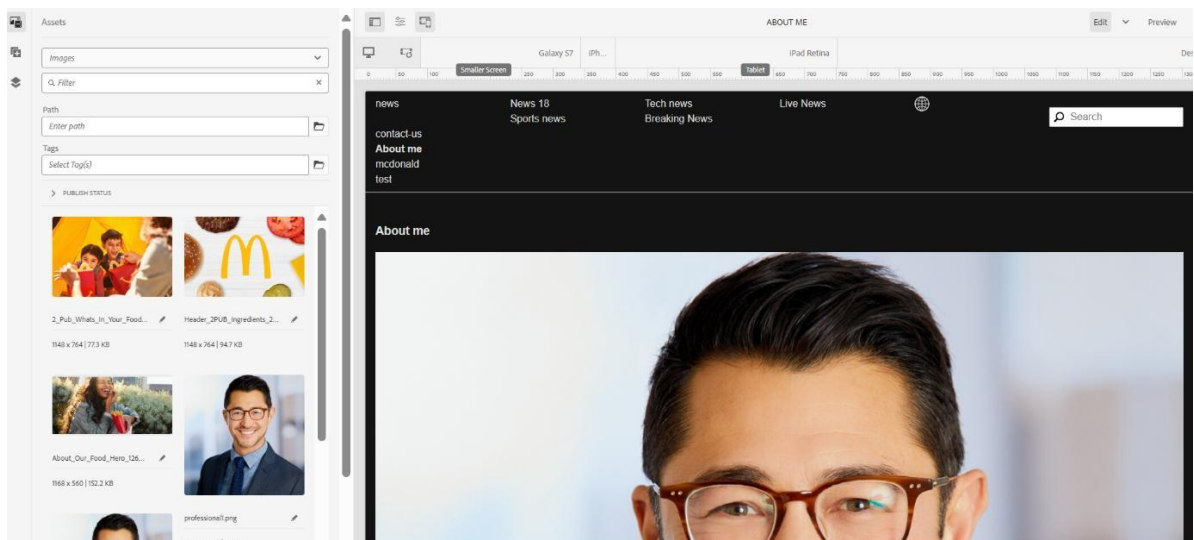
4. Save and publish.

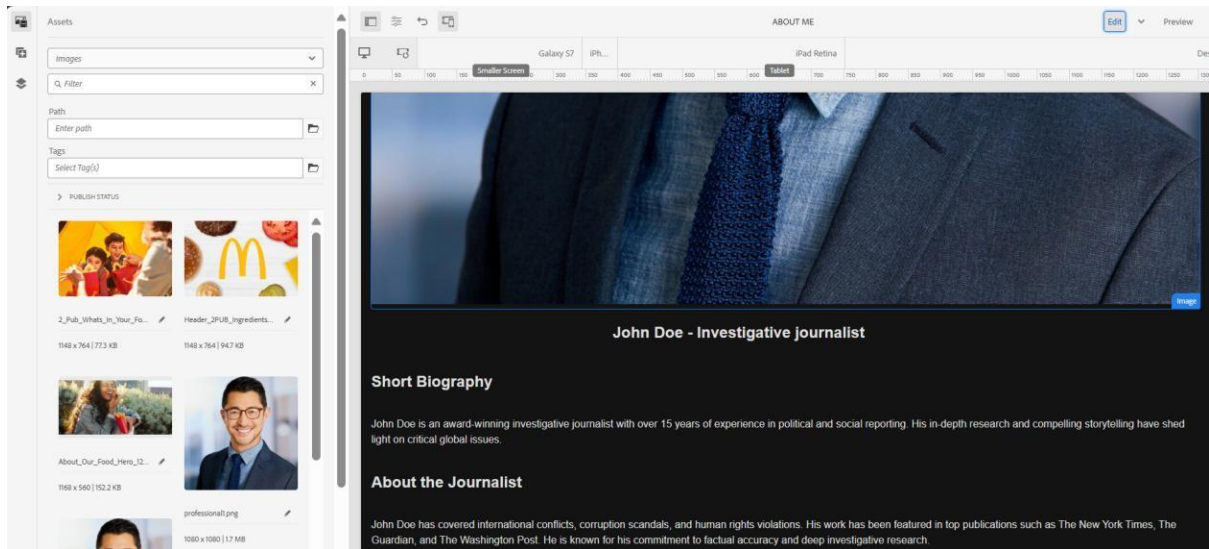
Screenshots

Contact page



About us page

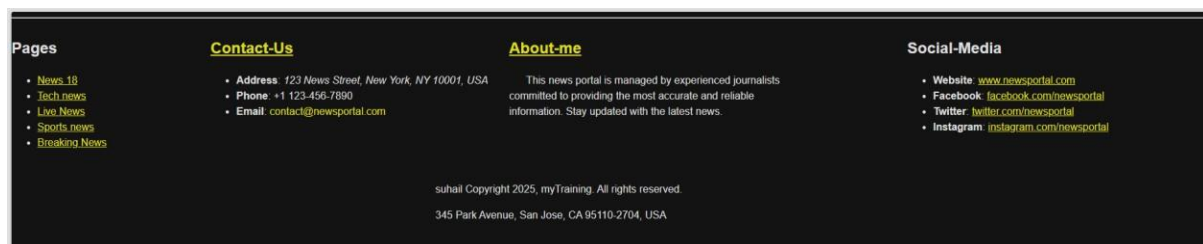




Step 4: Create Footer Experience Fragment (XF)

1. Navigate to AEM → Experience Fragments.
2. Click Create → Experience Fragment and choose Footer Template.
3. Name it News Footer, path: /content/experience-fragments/us/en/news-footer.
4. Add sections:
 - News Menu (List Component): Show 4 recent articles.
 - About Me (Text Component): Add journalist info.
 - Contact Us (Text Component): Add contact details.
 - Social Media (List Component): Add links to social profiles.
5. Save and publish.

Screenshot



Step 5: Create a Custom Service to Print "Hello World"

1. Navigate to your AEM Core Module and create a new Java service:

Java Code:

```
@Component(service = HelloWorldService.class, immediate = true)

public class HelloWorldService {

    public String getMessage() {

        return "Hello, World!";

    }

}
```

2. Inject this service into your News Component's Sling Model:

Java Code:

```
@Model(adaptables = Resource.class, defaultInjectionStrategy =
DefaultInjectionStrategy.OPTIONAL)

public class NewsModel {

    @OSGiService

    private HelloWorldService helloWorldService;

    public String getHelloMessage() {

        return helloWorldService.getMessage();

    }

}
```

3. Print the message in logs:

Java Code:

```
private static final Logger LOG = LoggerFactory.getLogger(NewsModel.class);

public String getHelloMessage() {

    String message = helloWorldService.getMessage();

    LOG.info("HelloWorldService Output: {}", message);

    return message;

}
```

4. Deploy and test it by checking AEM logs (tail -f error.log).

```
02.04.2025 07:31:33.044 *INFO* [[0:0:0:0:0:0:1] [1743559292456] GET /content/myTraining/News/jcr:content/root/container/container/news.html HTTP/1.1]
com.myTraining.core.models.NewsComponentModel News Component says: Hello World from AEM Service!
02.04.2025 07:31:33.058 *INFO* [[0:0:0:0:0:0:1] [1743559292984] GET /content/myTraining/News/jcr:content/root/container/container/news.html HTTP/1.1]
com.myTraining.core.models.NewsComponentModel News Component says: Hello World from AEM Service!
02.04.2025 07:31:38.394 *INFO* [[0:0:0:0:0:0:1] [1743559293069] GET /content/myTraining/us/en/news-1.html HTTP/1.1] com.myTraining.core.models.NewsComponentModel News Component says: Hello
World from AEM Service!
02.04.2025 07:32:00.807 *INFO* [[0:0:0:0:0:0:1] [1743559320193] GET /content/myTraining/us/en/news-1.html HTTP/1.1] com.myTraining.core.models.NewsComponentModel News Component says: Hello
World from AEM Service!
02.04.2025 07:32:04.929 *INFO* [[0:0:0:0:0:0:1] [1743559324401] GET /content/myTraining/us/en/news-4.html HTTP/1.1] com.myTraining.core.models.NewsComponentModel News Component says: Hello
World from AEM Service!
02.04.2025 07:32:09.585 *INFO* [[0:0:0:0:0:0:1] [1743559328919] GET /content/myTraining/us/en/hello.html HTTP/1.1] com.myTraining.core.models.NewsComponentModel News Component says: Hello
World from AEM Service!
02.04.2025 07:32:32.124 *INFO* [[0:0:0:0:0:0:1] [1743559352109] GET /content/myTraining/News/jcr:content/root/container/container/news.html HTTP/1.1]
com.myTraining.core.models.NewsComponentModel News Component says: Hello World from AEM Service!
```

Step 6: Create Custom Configurations for Third-Party API

1. Go to **AEM → Tools → Configuration Browser**.
2. Click **Create → Configuration** and name it **thirdPartyAPIConfig**.
3. Open **OSGi Configuration Console** and create a new configuration:

Java code:

```
@Component(service = ThirdPartyAPIService.class, immediate = true)
```

```
@Designate(ocd = ThirdPartyAPIConfig.class)
```

```
public class ThirdPartyAPIService {
```

```
    @Activate
```

```
    @Modified
```

```
    protected void activate(ThirdPartyAPIConfig config) {
```

```
        apiUrl = config.apiUrl();
```

```
    }
```

```
    public String fetchAPIData() {
```

```
        try (CloseableHttpClient client = HttpClients.createDefault()) {
```

```
            HttpGet request = new HttpGet(apiUrl);
```

```
            HttpResponse response = client.execute(request);
```

```
            return EntityUtils.toString(response.getEntity());
```

```
        } catch (Exception e) {
```

```
            LOG.error("Error fetching API data", e);
```

```
            return "{}";
```

```
        }
```



```
}
```

```
@ObjectClassDefinition(name = "Third-Party API Configuration")
```

```
public @interface ThirdPartyAPIConfig {
```

```
    @AttributeDefinition(name = "API URL", description = "Enter the API URL")
```

```
    String apiUrl() default "https://jsonplaceholder.typicode.com/posts";
```

```
}
```

4. Deploy and check logs (tail -f error.log)

- Expected Output: JSON data from <https://jsonplaceholder.typicode.com/posts>.

```
01.04.2025 19:34:36.011 *INFO* [OsgiInstallerImpl] com.mytraining.core.services.ApiService Fetching API data from: https://jsonplaceholder.typicode.com/posts
01.04.2025 19:34:37.138 *INFO* [OsgiInstallerImpl] com.mytraining.core.services.ApiService HTTP Response Code: 200
01.04.2025 19:34:37.162 *INFO* [OsgiInstallerImpl] com.mytraining.core.services.ApiService API Response Data: [ { "userId": 1, "id": 1, "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit", "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto" }, { "userId": 1, "id": 2, "title": "qui est esse", "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque nisi nulla" }, { "userId": 1, "id": 3, "title": "ea molestias quasi exercitationem repellat qui ipsa sit aut", "body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis pariatur\nmolestiae porro eius odio et labore et velit aut" }, { "userId": 1, "id": 4, "title": "eum et est occaecati", "body": "ullam et saepe reiciendis voluptatem adipisci\nsit amet autem assumenda provident rerum culpa\nquis hic commodi nesciunt rem tenetur doloremque ipsam iure\nquis sunt voluptatem rerum illo velit" }, { "userId": 1, "id": 5, "title": "nesciunt quas odio", "body": "repudiandae veniam quaerat sunt sed\nnalias aut fugiat sit autem sed est\nvoluptatem omnis possimus esse voluptatibus quis\nnest aut tenetur dolor neque" }, { "userId": 1, "id": 6, "title": "dolorem eum magni eos aperiam quia", "body": "ut aspernatur corporis harum nihil quis provident sequi\nmollitia nobis aliquid molestiae\nperspiciatis et ea nemo ab reprehenderit accusantium quas\nvoluptate dolores velit et doloremque molestiae" }, { "userId": 1, "id": 7, "title": "magnam facilis autem", "body": "dolore placeat quibusdam ea quo vitae\nmagni quis enim qui quis quo nemo aut saepe\nquidem repellat excepturi ut quia\nsunt ut sequi eos ea sed quas" }, { "userId": 1, "id": 8, "title": "dolorem dolore est ipsam", "body": "dignissimos aperiam dolorem qui eum\nfacilis quibusdam animi sint suscipit qui sint possimus cum\nquaerat magni maiores excepturi\nipsam ut commodi dolor voluptatum modi aut vitae" }, { "userId": 1, "id": 9, "title": "nesciunt iure omnis dolorem tempora et accusantium", "body": "consectetur animi nesciunt iure dolore\nnemin quia ad\nveniam autem ut quam aut nobis\net est aut quod aut provident voluptas autem voluptas" }, { "userId": 1, "id": 10, "title": "optio molestias id quia eum", "body": "quo et expedita modi cum officia vel magni\ndoloribus qui repudiandae\nvero nisi sit\nquos veniam quod sed accusamus veritatis error" }, { "userId": 2, "id": 11, "title": "et ea vero quia laudantium autem", "body": "delectus reiciendis molestiae occaecati non minima eveniet qui voluptatibus\naccusamus in eum beatae sit\nvel qui neque voluptates ut commodi qui incidunt\nut animi commodi" }, { "userId": 2, "id": 12, "title": "in quibusdam tempore odit est dolorem", "body": "itaque id aut magnam\npraesentium quia et ea odit et ea voluptas et\nsapiente quia nihil amet occaecati quia id voluptatem\nincidunt ea est distinctio odio" }, { "userId": 2, "id": 13, "title": "dolorum ut in voluptas mollitia et saepe quo animi", "body": "aut dicta possimus sint mollitia voluptas commodi quo doloremque\nniste corrupti reiciendis voluptatem eius rerum\nsit cumque quod eligendi laborum minima\nperferendis recusandae assumenda consectetur porro architecto ipsum ipsam" }, { "userId": 2, "id": 14, "title": "voluptatem eligendi optio", "body": "fuga et accusamus dolorum perferendis illo voluptas\nnon doloremque neque facere\nnadi qui dolorum molestiae beatae\nsed aut voluptas totam sit illum" }, { "userId": 2, "id": 15, "title": "eveniet quod temporibus", "body": "reprehenderit quos placeat\nvelit minima officia dolores impedit repudiandae molestiae nam\nvoluptas recusandae quis delectus\nnofficiis harum fugiat vitae" }, { "userId": 2, "id": 16, "title": "sint suscipit perspiciatis velit", "body": "dolorum rerum ipsa laboriosam odio", "body": "suscipit nam nisi quo aperiam aut\nasperiores eos fugit maiores voluptatibus quia\nvoluptatem quis ullam qui in alias quia est\nconsequatur magni mollitia accusamus ea nisi voluptate dicta" }, { "userId": 2, "id": 17, "title": "fugit voluptas sed molestias voluptatem provident", "body": "eos voluptas et aut odit natus earum\naspernatur fuga molestiae ullam\nvdseserunt ratione qui eos\nqui nihil ratione nemo velit ut aut id quo" }, { "userId": 2, "id": 18, "title": "voluptate et itaque vero tempora molestiae", "body": "eveniet quo quis\nlaborum totam consequatur non dolor\nut et est repudiandae\nnest voluptatem vel debitis et magnam" }, { "userId": 2, "id": 19, "title": "adipisci placeat illum aut reiciendis qui", "body": "illum quis cupiditate provident sit magnam\nnea sed aut omnis\nveniam maiores ullam consequatur atque\nnadi ipsi qui iste expedita sit quos voluptas" }, { "userId": 2, "id": 20, "title": "doloribus ad provident suscipit at", "body": "qui consequuntur ducimus possimus quisquam amet similique\nsuscipit porro ipsam amet\nneos veritatis officiis exercitationem vel fugit aut necessitatibus totam\nnomnis rerum consequatur expedita quidem cumque explicabo" }, { "userId": 3, "id": 21, "title": "asperiores ea ipsam voluptatibus modi minima quia sint", "body": "repellat aliquid praesentium dolorem quo\nsed totam minus non itaque\nnihil labore molestiae sunt dolor eveniet hic recusandae veniam\ntempora et tenetur expedita sunt" }, { "userId": 3, "id": 22, "title": "dolor sint quo a velit explicabo quia nam", "body": "eos qui et ipsum ipsam suscipit aut\nsed omnis non odio\nexpedita earum mollitia molestiae aut atque rem suscipit\nnam impedit esse" }, { "userId": 3, "id": 23, "title": "maxime id vitae nihil numquam", "body": "veritatis unde neque eligendi\nquae quod architecto quo neque vitae\nnest illo sit tempora doloremque fugit quod\nnet et vel beatae sequi ullam sed tenetur perspiciatis" }, { "userId": 3, "id": 24, "title": "autem hic labore sunt dolores incidunt", "body": "enim et ex nulla\nnomnis voluptas quia qui\nvoluptatem consequatur numquam aliquam sunt\ntotam recusandae id dignissimos aut sed asperiores deserunt" }, { "userId": 3, "id": 25, "title": "rem alias distinctio quo"}]
```