Machine Learning Model Deployment with IBM Cloud Watson Studio

INTRODUCTION:

Machine Learning Model Deployment with IBM Cloud Watson Studio is a process that involves taking a trained machine learning model and making it accessible to applications or services. IBM Cloud Watson Studio provides a platform for deploying, managing, and scaling machine learning models, allowing you to harness the power of AI in your applications. It offers tools for model deployment, monitoring, and integration, making it easier to put your AI models into production and leverage their predictive capabilities in real-world scenarios.

CODING:

```
import com.ibm.cloud.sdk.core.security.IamAuthenticator;
import com.ibm.watson.machine_learning.v4.WatsonMachineLearning;
import com.ibm.watson.machine_learning.v4.model.*;
import java.io.File;

public class ModelDeploymentExample {

    public static void main(String[] args) {
        // Set up the authentication
        String apiKey = "YOUR_API_KEY";
        IamAuthenticator authenticator = new IamAuthenticator(apiKey);

        // Create an instance of WatsonMachineLearning
        WatsonMachineLearning wmlClient = new WatsonMachineLearning(authenticator);

        // Set the deployment space ID
        String deploymentSpaceId = "YOUR_DEPLOYMENT_SPACE_ID";

        // Set the model details
        String modelName = "YOUR_MODEL_NAME";
        String modelFilePath = "PATH_TO_MODEL_FILE";

        try {
            // Load the model file
            File modelFile = new File(modelFilePath);
```

```
        // Create a new model artifact
        CreateModelOptions createModelOptions = new CreateModelOptions.Builder()
                .name(modelName)
                .spaceId(deploymentSpaceId)
                .addFiles(modelFile)
                .build();

        // Create the model in Watson Studio
        Model createModelResponse =
wmlClient.createModel(createModelOptions).execute().getResult();

        // Get the model ID
        String modelId = createModelResponse.getModelId();

        // Define the deployment metadata
        DeploymentMetadata deploymentMetadata = new DeploymentMetadata.Builder()
                .name("YOUR_DEPLOYMENT_NAME")
                .build();

        // Define the deployment resource requirements
        DeploymentResourceRequirements deploymentResourceRequirements =
                new DeploymentResourceRequirements.Builder()
                        .build();

        // Create a new deployment
```

```
            CreateDeploymentOptions createDeploymentOptions = new
CreateDeploymentOptions.Builder()
                    .name("YOUR_DEPLOYMENT_NAME")
                    .spaceId(deploymentSpaceId)
                    .addAsset(modelId)
                    .metadata(deploymentMetadata)
                    .resourceRequirements(deploymentResourceRequirements)
                    .build();

            // Deploy the model
            Deployment createDeploymentResponse =
wmlClient.createDeployment(createDeploymentOptions).execute().getResult();

            // Get the deployment ID
            String deploymentId = createDeploymentResponse.getDeploymentId();

            System.out.println("Model deployed successfully with deployment ID: " + deploymentId);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```