Task 1: Data Preprocessing and Feature Engineering
Objective:
Clean the dataset and prepare features for ML algorithms.

Steps:
1. Fill missing values in TotalCharges with 0.
2. Encode categorical features using StringIndexer and OneHotEncoder.
3. Assemble numeric and encoded features into a single feature vector with VectorAssembler.

Code Output:

```
+-----------------------------------------------------+----------+
|features                                             |ChurnIndex|
+-----------------------------------------------------+----------+
|(11,[1,2,3,5,7,9],[7.0,62.85,443.15,1.0,1.0,1.0])    |1         |
|(11,[1,2,3,4,6,8],[62.0,97.45,6186.07,1.0,1.0,1.0]) |0         |
|(11,[1,2,3,5,7,9],[5.0,94.68,446.96,1.0,1.0,1.0])    |0         |
|[1.0,6.0,87.75,574.8,1.0,0.0,1.0,0.0,1.0,0.0,0.0]    |1         |
|(11,[1,2,3,5,7,10],[61.0,45.22,2188.85,1.0,1.0,1.0])|0         |
+-----------------------------------------------------+----------+
only showing top 5 rows
```

Task 2: Train and Evaluate Logistic Regression Model
Objective:
Train a logistic regression model and evaluate it using AUC (Area Under ROC Curve).

Steps:
1. Split dataset into training and test sets (80/20).
2. Train a logistic regression model.
3. Use BinaryClassificationEvaluator to evaluate.

Code Output Example:

```
25/04/08 17:14:02 WARN InstanceBuilder: Failed to load implementation from:dev.ludovic.netlib.blas.JNIBLAS
25/04/08 17:14:02 WARN InstanceBuilder: Failed to load implementation from:dev.ludovic.netlib.blas.VectorBLAS
25/04/08 17:14:03 WARN GarbageCollectionMetrics: To enable non-built-in garbage collector(s) List(G1 Concurrent GC), users should configure it(them) to spark.eventLog.gcMetrics.youngGenerationGarbageCollectors or spark.eventLog.gcMetrics.oldGenerationGarbageCollectors
Logistic Regression Model Accuracy (AUC): 0.80
```

Task 3: Feature Selection using Chi-Square Test
Objective:
Select the top 5 most important features using Chi-Square feature selection.

Steps:
1. Use ChiSqSelector to rank and select top 5 features.
2. Print the selected feature vectors.

Code Output Example:

```
Selected features after ChiSqSelector:
+-------------------------+----------+
|selectedFeatures         |ChurnIndex|
+-------------------------+----------+
|(5,[1,4],[7.0,1.0])      |1         |
|(5,[1,2,3],[62.0,1.0,1.0])|0         |
|(5,[1,4],[5.0,1.0])      |0         |
|[1.0,6.0,1.0,1.0,0.0]    |1         |
|(5,[1],[61.0])           |0         |
+-------------------------+----------+
only showing top 5 rows
```

```
Task 4: Hyperparameter Tuning and Model Comparison
Objective:
Use CrossValidator to tune models and compare their AUC performance.

Models Used:
  - Logistic Regression
  - Decision Tree Classifier
  - Random Forest Classifier
  - Gradient Boosted Trees (GBT)

Steps:
  1. Define models and parameter grids.
  2. Use CrossValidator for 5-fold cross-validation.
  3. Evaluate and print best model results.

Code Output Example:
Tuning LogisticRegression...
LogisticRegression Best Model Accuracy (AUC): 0.80
Best Params for LogisticRegression: {Param(parent='LogisticRegression_36b94b251ce3', name='aggregationDepth', doc='suggested depth for treeAggregate (>= 2).'): 2, Param(parent='LogisticRegression_36b94b251ce3', name='elasticNetParam', doc='the ElasticNet mixing parameter, in range [0, 1]. For alpha = 0, the penalty is an L2 penalty. For alpha = 1, it is an L1 penalty.'): 0.0, Param(parent='LogisticRegression_36b94b251ce3', name='family', doc='the name of family which is a description of the label distribution to be used in the model. Supported options: auto, binomial, multinomial'): 'auto', Param(parent='LogisticRegression_36b94b251ce3', name='featuresCol', doc='features column name.'): 'features', Param(parent='LogisticRegression_36b94b251ce3', name='fitIntercept', doc='whether to fit an intercept term.'): True, Param(parent='LogisticRegression_36b94b251ce3', name='labelCol', doc='label column name.'): 'ChurnIndex', Param(parent='LogisticRegression_36b94b251ce3', name='maxBlockSizeInMB', doc='maximum memory in MB for stacking input data into blocks. Data is stacked within partitions. If more than remaining data size in a partition then it is adjusted to the data size. Default 0.0 represents choosing optimal value, depends on specific algorithm. Must be >= 0.'): 0.0, Param(parent='LogisticRegression_36b94b251ce3', name='maxIter', doc='maximum number of iterations (>= 0).'): 10, Param(parent='LogisticRegression_36b94b251ce3', name='predictionCol', doc='prediction column name.'): 'prediction', Param(parent='LogisticRegression_36b94b251ce3', name='probabilityCol', doc='Column name for predicted class conditional probabilities. Note: Not all models output well-calibrated probability estimates! These probabilities should be treated as confidences, not precise probabilities.'): 'probability', Param(parent='LogisticRegression_36b94b251ce3', name='rawPredictionCol', doc='raw prediction (a.k.a. confidence) column name.'): 'rawPrediction', Param(parent='LogisticRegression_36b94b251ce3', name='regParam', doc='regularization parameter (>= 0).'): 0.01, Param(parent='LogisticRegression_36b94b251ce3', name='standardization', doc='whether to standardize the training features before fitting the model.'): True, Param(parent='LogisticRegression_36b94b251ce3', name='threshold', doc='threshold in binary classification prediction, in range [0, 1]. If threshold and thresholds are both set, they must match.e.g. if threshold is p, then thresholds must be equal to [1-p, p].'): 0.5, Param(parent='LogisticRegression_36b94b251ce3', name='tol', doc='the convergence tolerance for iterative algorithms (>= 0).'): 1e-06}
```

```
Tuning DecisionTree...
DecisionTree Best Model Accuracy (AUC): 0.78
Best Params for DecisionTree: {Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='cacheNodeIds', doc='If false, the algorithm will pass trees to executors to match instances with nodes. If true, the algorithm will cache node IDs for each instance. Caching can speed up training of deeper trees. Users can set how often should the cache be checkpointed or disable it by setting checkpointInterval.'): False, Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='checkpointInterval', doc='set checkpoint interval (>= 1) or disable checkpoint (-1). E.g. 10 means that the cache will get checkpointed every 10 iterations. Note: this setting will be ignored if the checkpoint directory is not set in the SparkContext.'): 10, Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='featuresCol', doc='features column name.'): 'features', Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='impurity', doc='Criterion used for information gain calculation (case-insensitive). Supported options: entropy, gini'): 'gini', Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='labelCol', doc='label column name.'): 'ChurnIndex', Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='leafCol', doc='Leaf indices column name. Predicted leaf index of each instance in each tree by preorder.'): '', Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='maxBins', doc='Max number of bins for discretizing continuous features.  Must be >=2 and >= number of categories for any categorical feature.'): 32, Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='maxDepth', doc='Maximum depth of the tree. (>= 0) E.g., depth 0 means 1 leaf node; depth 1 means 1 internal node + 2 leaf nodes. Must be in range [0, 30].'): 5, Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='maxMemoryInMB', doc='Maximum memory in MB allocated to histogram aggregation. If too small, then 1 node will be split per iteration, and its aggregates may exceed this size.'): 256, Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='minInfoGain', doc='Minimum information gain for a split to be considered at a tree node.'): 0.0, Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='minInstancesPerNode', doc='Minimum number of instances each child must have after split. If a split causes the left or right child to have fewer than minInstancesPerNode, the split will be discarded as invalid. Should be >= 1.'): 2, Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='minWeightFractionPerNode', doc='Minimum fraction of the weighted sample count that each child must have after split. If a split causes the fraction of the total weight in the left or right child to be less than minWeightFractionPerNode, the split will be discarded as invalid. Should be in interval [0.0, 0.5).'): 0.0, Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='predictionCol', doc='prediction column name.'): 'prediction', Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='probabilityCol', doc='Column name for predicted class conditional probabilities. Note: Not all models output well-calibrated probability estimates! These probabilities should be treated as confidences, not precise probabilities.'): 'probability', Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='rawPredictionCol', doc='raw prediction (a.k.a. confidence) column name.'): 'rawPrediction', Param(parent='DecisionTreeClassifier_e1c0e37a4e98', name='seed', doc='random seed.'): -3973981264758300543}
```

```
RandomForest Best Model Accuracy (AUC): 0.87
Best Params for RandomForest: {Param(parent='RandomForestClassifier_97ad0cf34663', name='bootstrap', doc='Whether bootstrap samples are used when building trees.'): True, Param(parent='RandomForestClassifier_97ad0cf34663', name='cacheNodeIds', doc='If false, the algorithm will pass trees to executors to match instances with nodes. If true, the algorithm will cache node IDs for each instance. Caching can speed up training of deeper trees. Users can set how often should the cache be checkpointed or disable it by setting checkpointInterval.'): False, Param(parent='RandomForestClassifier_97ad0cf34663', name='checkpointInterval', doc='set checkpoint interval (>= 1) or disable checkpoint (-1). E.g. 10 means that the cache will get checkpointed every 10 iterations. Note: this setting will be ignored if the checkpoint directory is not set in the SparkContext.'): 10, Param(parent='RandomForestClassifier_97ad0cf34663', name='featureSubsetStrategy', doc="The number of features to consider for splits at each tree node. Supported options: 'auto' (choose automatically for task: If numTrees == 1, set to 'all'. If numTrees > 1 (forest), set to 'sqrt' for classification and to 'onethird' for regression), 'all' (use all features), 'onethird' (use 1/3 of the features), 'sqrt' (use sqrt(number of features)), 'log2' (use log2(number of features)), 'n' (when n is in the range (0, 1.0], use n * number of features. When n is in the range (1, number of features), use n features). default = 'auto'"): 'auto', Param(parent='RandomForestClassifier_97ad0cf34663', name='featuresCol', doc='features column name.'): 'features', Param(parent='RandomForestClassifier_97ad0cf34663', name='impurity', doc='Criterion used for information gain calculation (case-insensitive). Supported options: entropy, gini'): 'gini', Param(parent='RandomForestClassifier_97ad0cf34663', name='labelCol', doc='label column name.'): 'ChurnIndex', Param(parent='RandomForestClassifier_97ad0cf34663', name='leafCol', doc='Leaf indices column name. Predicted leaf index of each instance in each tree by preorder.'): '', Param(parent='RandomForestClassifier_97ad0cf34663', name='maxBins', doc='Max number of bins for discretizing continuous features.  Must be >=2 and >= number of categories for any categorical feature.'): 32, Param(parent='RandomForestClassifier_97ad0cf34663', name='maxDepth', doc='Maximum depth of the tree. (>= 0) E.g., depth 0 means 1 leaf node; depth 1 means 1 internal node + 2 leaf nodes. Must be in range [0, 30].'): 5, Param(parent='RandomForestClassifier_97ad0cf34663', name='maxMemoryInMB', doc='Maximum memory in MB allocated to histogram aggregation. If too small, then 1 node will be split per iteration, and its aggregates may exceed this size.'): 256, Param(parent='RandomForestClassifier_97ad0cf34663', name='minInfoGain', doc='Minimum information gain for a split to be considered at a tree node.'): 0.0, Param(parent='RandomForestClassifier_97ad0cf34663', name='minInstancesPerNode', doc='Minimum number of instances each child must have after split. If a split causes the left or right child to have fewer than minInstancesPerNode, the split will be discarded as invalid. Should be >= 1.'): 1, Param(parent='RandomForestClassifier_97ad0cf34663', name='minWeightFractionPerNode', doc='Minimum fraction of the weighted sample count that each child must have after split. If a split causes the fraction of the total weight in the left or right child to be less than minWeightFractionPerNode, the split will be discarded as invalid. Should be in interval [0.0, 0.5).'): 0.0, Param(parent='RandomForestClassifier_97ad0cf34663', name='numTrees', doc='Number of trees to train (>= 1).'): 50, Param(parent='RandomForestClassifier_97ad0cf34663', name='predictionCol', doc='prediction column name.'): 'prediction', Param(parent='RandomForestClassifier_97ad0cf34663', name='probabilityCol', doc='Column name for predicted class conditional probabilities. Note: Not all models output well-calibrated probability estimates! These probabilities should be treated as confidences, not precise probabilities.'): 'probability', Param(parent='RandomForestClassifier_97ad0cf34663', name='rawPredictionCol', doc='raw prediction (a.k.a. confidence) column name.'): 'rawPrediction', Param(parent='RandomForestClassifier_97ad0cf34663', name='seed', doc='random seed.'): 7631400570407998011, Param(parent='RandomForestClassifier_97ad0cf34663', name='subsamplingRate', doc='Fraction of the training data used for learning each decision tree, in range (0, 1].'): 1.0}

Tuning GBT...
GBT Best Model Accuracy (AUC): 0.85
Best Params for GBT: {Param(parent='GBTClassifier_be38a1a4ca7a', name='cacheNodeIds', doc='If false, the algorithm will pass trees to executors to match instances with nodes. If true, the algorithm will cache node IDs for each instance. Caching can speed up training of deeper trees. Users can set how often should the cache be checkpointed or disable it by setting checkpointInterval.'): False, Param(parent='GBTClassifier_be38a1a4ca7a', name='checkpointInterval', doc='set checkpoint interval (>= 1) or disable checkpoint (-1). E.g. 10 means that the cache will get checkpointed every 10 iterations. Note: this setting will be ignored if the checkpoint directory is not set in the SparkContext.'): 10, Param(parent='GBTClassifier_be38a1a4ca7a', name='featureSubsetStrategy', doc="The number of features to consider for splits at each tree node. Supported options: 'auto' (choose automatically for task: If numTrees == 1, set to 'all'. If numTrees > 1 (forest), set to 'sqrt' for classification and to 'onethird' for regression), 'all' (use all features), 'onethird' (use 1/3 of the features), 'sqrt' (use sqrt(number of features)), 'log2' (use log2(number of features)), 'n' (when n is in the range (0, 1.0], use n * number of features. When n is in the range (1, number of features), use n features). default = 'auto'"): 'all', Param(parent='GBTClassifier_be38a1a4ca7a', name='featuresCol', doc='features column name.'): 'features', Param(parent='GBTClassifier_be38a1a4ca7a', name='impurity', doc='Criterion used for information gain calculation (case-insen
```