Project 3: **Predicting Equipment Pricing Using Spark**

For this project we were given a dataset containing 300,000 records of a heavy equipment auction. These auctions were described using 53 different variables, where the variable of interest was the final sale price for each unique piece of equipment. PySpark was used due to its faster computation time, primarily due to its in-memory caching of data. Although this dataset is not particularly large, it was extremely valuable to practice this increasingly popular tool in data analyses.

The model used for predicting heavy equipment auction sales price was linear regression with a regularization term (0.8). The first task was selecting the independent variables for this model. Variables with at least half of its values being null were not considered, and neither were categorical ones with over 80 factor levels. With reference to provided data dictionary, variables that were very similar to one another were also removed to prevent any multicollinearity.

This criterion directed to 6 independent variables used in the model: *YearMade*, *ProductGroup*, *Drive_System*, *Enclosure*, *Forks*, and *Pushblock*. Bar *YearMade* (numeric), all these variables were categorical. Categorical variables can be difficult to implement in pySpark since strings are not supported when training many ML models on the framework. To implement each categorical variable, a pySpark function called StringIndexer was used to map a string to an index value. These indices were mapped in terms of frequency, string values that occurred more often would be given a lower index number. A Pipeline was used to run the transform() and fit() methods in the proper order since multiple categorical variable were delt with.

After all categorical variables values were mapped to an index, they needed to go through one final transformation. VectorAssembler, a pySpark ML feature, was used to merge each observation's independent variable value into a feature vector. The final training data set would match each feature vector with its corresponding sale price. The trained model produced the following coefficients:

Trained Model Coefficients	
Variable	Coefficient
YearMade	12.38
Product_Group (Index)	203.28
Drive_System (Index)	-1710.03
Enclosure (Index)	4533.02
Forks (Index)	4383.59
Pushblock (Index)	2845.80

In comparison to processing time using R, this model fitting was almost instantaneous. Although a lot more data preprocessing was required for pySpark, when using large datasets this con is outweighed by its computational strengths. Refer to the "Test_pred.csv" file for the predicted auction sale for each row.