using System;

using System.Linq;

namespace TradingStrategy

{

class Program

{

static void Main(string[] args)

{

// Initialize a list of stocks to evaluate

var stocks = new Stock[]

{

new Stock("Apple", 120.00, 8.0, 0.1, 0.12, 0.05, 0.06, 0.07),

new Stock("Microsoft", 100.00, 7.0, 0.2, 0.15, 0.04, 0.05, 0.06),

new Stock("Google", 150.00, 9.0, 0.3, 0.18, 0.06, 0.07, 0.08)

};

// Evaluate each stock based on the trading strategy

foreach (var stock in stocks)

{

Console.WriteLine("Evaluating stock: " + stock.Name);

EvaluateStock(stock);

Console.WriteLine("");

}

// Wait for user input to exit the program

Console.ReadLine();

}

static void EvaluateStock(Stock stock)

{

// Calculate the score for each factor

var earningsGrowthScore = EvaluateEarningsGrowth(stock.EarningsGrowth);

var debtToEquityScore = EvaluateDebtToEquity(stock.DebtToEquity);

var roeScore = EvaluateROE(stock.ROE);

var industryTrendsScore = EvaluateIndustryTrends(stock.IndustryTrends);

var managementScore = EvaluateManagement(stock.Management);

var marketConditionsScore = EvaluateMarketConditions(stock.MarketConditions);

var riskToleranceScore = EvaluateRiskTolerance(stock.RiskTolerance);

var p\_eScore = EvaluatePE(stock.PE);

var p\_bScore = EvaluatePB(stock.PB);

var dividendYieldScore = EvaluateDividendYield(stock.DividendYield);

var marketTrendsScore = EvaluateMarketTrends(stock.MarketTrends);

var management\_Score = EvaluateManagement(stock.Management);

var competitiveEnvironmentScore = EvaluateCompetitiveEnvironment(stock.CompetitiveEnvironment);

var financialsScore = EvaluateFinancials(stock.Financials);

var diversificationScore = EvaluateDiversification(stock.Diversification);

var timingScore = EvaluateTiming(stock.Timing);

var taxConsiderationsScore = EvaluateTaxConsiderations(stock.TaxConsiderations);

// Calculate the overall score

var overallScore = (earningsGrowthScore + debtToEquityScore + roeScore + industryTrendsScore +

managementScore + marketConditionsScore + riskToleranceScore + p\_eScore + p\_bScore +

dividendYieldScore + marketTrendsScore + management\_Score + competitiveEnvironmentScore +

financialsScore + diversificationScore + timingScore + taxConsiderationsScore) / 15;

// Determine if the stock is a good investment based on the overall score

if (overallScore >= 0.7)

{

Console.WriteLine("Stock