//@version=4  
  
strategy("RSI Divergence", shorttitle = "RSI-DIV", overlay=false, pyramiding=2, commission\_type=strategy.commission.percent, commission\_value=0.07, default\_qty\_type = strategy.percent\_of\_equity, default\_qty\_value = 100, currency = currency.USD )  
  
ribbon\_period = input(15, "Period", step=1)  
  
leadLine1 = ema(close, ribbon\_period)  
leadLine2 = sma(close, ribbon\_period)  
  
// p1 = plot(leadLine1, color= #53b987, title="EMA", transp = 50, linewidth = 1)  
// p2 = plot(leadLine2, color= #eb4d5c, title="SMA", transp = 50, linewidth = 1)  
// fill(p1, p2, transp = 60, color = leadLine1 > leadLine2 ? #53b987 : #eb4d5c)  
  
  
len = input(title="RSI Period", minval=1, defval=7)  
src = input(title="RSI Source", defval=close)  
lbR = input(title="Pivot Lookback Right", defval=3)  
lbL = input(title="Pivot Lookback Left", defval=4)  
takeProfitRSILevel\_long = input(title="Long Take Profit at RSI Level", minval=60, defval=77)  
takeProfitRSILevel\_short=input(title="Short Take Profit at RSI Level", maxval=30, defval=19)  
  
rangeUpper = input(title="Max of Lookback Range", defval=19)  
rangeLower = input(title="Min of Lookback Range", defval=6)  
plotBull = input(title="Plot Bullish", defval=true)  
plotHiddenBull = input(title="Plot Hidden Bullish", defval=true)  
plotBear = input(title="Plot Bearish", defval=true)  
plotHiddenBear = input(title="Plot Hidden Bearish", defval=false)  
  
  
bearColor = color.purple  
bullColor = color.green  
hiddenBullColor = color.new(color.green, 80)  
hiddenBearColor = color.new(color.red, 80)  
textColor = color.white  
noneColor = color.new(color.white, 100)  
  
osc = rsi(src, len)  
  
plot(osc, title="RSI", linewidth=2, color=#8D1699)  
hline(50, title="Middle Line", linestyle=hline.style\_dotted)  
obLevel = hline(70, title="Overbought", linestyle=hline.style\_dotted)  
osLevel = hline(30, title="Oversold", linestyle=hline.style\_dotted)  
fill(obLevel, osLevel, title="Background", color=#9915FF, transp=90)  
  
plFound = na(pivotlow(osc, lbL, lbR)) ? false : true  
phFound = na(pivothigh(osc, lbL, lbR)) ? false : true  
  
\_inRange(cond) =>  
 bars = barssince(cond == true)  
 rangeLower <= bars and bars <= rangeUpper  
  
//------------------------------------------------------------------------------  
// Regular Bullish  
  
// Osc: Higher Low  
oscHL = osc[lbR] > valuewhen(plFound, osc[lbR], 1) and \_inRange(plFound[1])  
  
// Price: Lower Low  
priceLL = low[lbR] < valuewhen(plFound, low[lbR], 1)  
  
bullCond = plotBull and priceLL and oscHL and plFound  
  
plot(  
 plFound ? osc[lbR] : na,  
 offset=-lbR,  
 title="Regular Bullish",  
 linewidth=2,  
 color=(bullCond ? bullColor : noneColor),  
 transp=0  
 )  
  
  
plotshape(  
 bullCond ? osc[lbR] : na,  
 offset=-lbR,  
 title="Regular Bullish Label",  
 text=" Bull ",  
 style=shape.labelup,  
 location=location.absolute,  
 color=bullColor,  
 textcolor=textColor,  
 transp=0  
 )  
  
//------------------------------------------------------------------------------  
// Hidden Bullish  
  
// Osc: Lower Low  
oscLL = osc[lbR] < valuewhen(plFound, osc[lbR], 1) and \_inRange(plFound[1])  
  
// Price: Higher Low  
priceHL = low[lbR] > valuewhen(plFound, low[lbR], 1)  
  
hiddenBullCond = plotHiddenBull and priceHL and oscLL and plFound  
  
plot(  
 plFound ? osc[lbR] : na,  
 offset=-lbR,  
 title="Hidden Bullish",  
 linewidth=2,  
 color=(hiddenBullCond ? hiddenBullColor : noneColor),  
 transp=0  
 )  
  
plotshape(  
 hiddenBullCond ? osc[lbR] : na,  
 offset=-lbR,  
 title="Hidden Bullish Label",  
 text=" H Bull ",  
 style=shape.labelup,  
 location=location.absolute,  
 color=bullColor,  
 textcolor=textColor,  
 transp=0  
 )  
  
//------------------------------------------------------------------------------  
// Regular Bearish  
  
// Osc: Lower High  
oscLH = osc[lbR] < valuewhen(phFound, osc[lbR], 1) and \_inRange(phFound[1])  
  
// Price: Higher High  
priceHH = high[lbR] > valuewhen(phFound, high[lbR], 1)  
  
bearCond = plotBear and priceHH and oscLH and phFound  
  
plot(  
 phFound ? osc[lbR] : na,  
 offset=-lbR,  
 title="Regular Bearish",  
 linewidth=2,  
 color=(bearCond ? bearColor : noneColor),  
 transp=0  
 )  
  
plotshape(  
 bearCond ? osc[lbR] : na,  
 offset=-lbR,  
 title="Regular Bearish Label",  
 text=" Bear ",  
 style=shape.labeldown,  
 location=location.absolute,  
 color=bearColor,  
 textcolor=textColor,  
 transp=0  
 )  
  
//------------------------------------------------------------------------------  
// Hidden Bearish  
  
// Osc: Higher High  
oscHH = osc[lbR] > valuewhen(phFound, osc[lbR], 1) and \_inRange(phFound[1])  
  
// Price: Lower High  
priceLH = high[lbR] < valuewhen(phFound, high[lbR], 1)  
  
hiddenBearCond = plotHiddenBear and priceLH and oscHH and phFound  
  
plot(  
 phFound ? osc[lbR] : na,  
 offset=-lbR,  
 title="Hidden Bearish",  
 linewidth=2,  
 color=(hiddenBearCond ? hiddenBearColor : noneColor),  
 transp=0  
 )  
  
plotshape(  
 hiddenBearCond ? osc[lbR] : na,  
 offset=-lbR,  
 title="Hidden Bearish Label",  
 text=" H Bear ",  
 style=shape.labeldown,  
 location=location.absolute,  
 color=bearColor,  
 textcolor=textColor,  
 transp=0  
 )  
  
  
/// Strategy Conditions  
  
  
entry\_long = if leadLine2<leadLine1  
 hiddenBullCond  
else  
 bullCond  
  
entry\_price\_long=valuewhen(entry\_long,close,0)  
exit\_long = crossover(osc,takeProfitRSILevel\_long)// or bearCond  
  
  
///// SHORT ////  
  
entry\_short = if leadLine2>leadLine1  
 hiddenBearCond  
else  
 bearCond  
  
entry\_price\_short=valuewhen(entry\_short,close,0)  
exit\_short = crossunder(osc,takeProfitRSILevel\_short)// or bullCond  
  
  
///// BACKTEST PERIOD ///////  
testStartYear = input(2019, "Backtest Start Year")  
testStartMonth = input(1, "Backtest Start Month")  
testStartDay = input(1, "Backtest Start Day")  
testPeriodStart = timestamp(testStartYear,testStartMonth,testStartDay,0,0)  
  
testStopYear = input(9999, "Backtest Stop Year")  
testStopMonth = input(12, "Backtest Stop Month")  
testStopDay = input(31, "Backtest Stop Day")  
testPeriodStop = timestamp(testStopYear,testStopMonth,testStopDay,0,0)  
  
testPeriod() =>  
 time >= testPeriodStart and time <= testPeriodStop ? true : false  
  
if testPeriod()  
 if strategy.position\_size == 0 or strategy.position\_size > 0  
 strategy.entry(id="long", long=true, when = entry\_long, comment="ENTER-LONG\_BINANCE-FUTURES\_ETHUSDT\_buysell-test\_15M\_b95d558576cf96d4")  
 strategy.close(id="long", when=exit\_long, comment = "EXIT-LONG\_BINANCE-FUTURES\_ETHUSDT\_buysell-test\_15M\_b95d558576cf96d4")  
  
 if strategy.position\_size == 0 or strategy.position\_size < 0  
 strategy.entry(id="short", long=false, when = entry\_short, comment="ENTER-SHORT\_BINANCE-FUTURES\_ETHUSDT\_buysell-test\_15M\_b95d558576cf96d4")  
 strategy.close(id="short", when=exit\_short, comment = "EXIT-SHORT\_BINANCE-FUTURES\_ETHUSDT\_buysell-test\_15M\_b95d558576cf96d4")