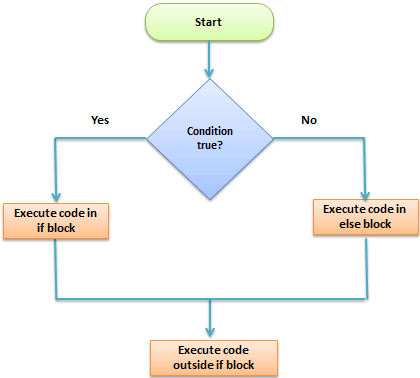
Control Structures:

Commonly used control structures are

1. if and else: testing a condition and acting on it
2. for: execute a loop a fixed number of times
3. while: execute a loop while a condition is true
4. repeat: execute an infinite loop (must break out of it to stop)
5. break: break the execution of a loop
6. next: skip an interation of a loop

IF ELSE:



General Structure

if ( test\_expression1) {

statement1

} else if ( test\_expression2) {

statement2

} else if ( test\_expression3) {

statement3

} else {

statement4

}

#Hotel Example

data$Leader\_flag <- ifelse(data$Role=='Chef', "Head of Department", "Normal Employee")

data

data$spec\_grp <-

with(data,

ifelse(data$Role=="Chef","Rank 1",

ifelse(data$Role=="Cook","Rank 2",

ifelse(data$Role=="Cashier", "Rank 3",

ifelse(data$Role=="Manager", "Rank 4",

ifelse(data$Role=="Sales Executive", "Rank 5","Rank 6"))))))

df3 <- data.frame(x=c("A","A","B","A"), y=c("A","B","B","B"))

df3

df3$z <- ifelse(df3$x == "A" & df3$y == "A", "NA", "B")

df3

CASE (If else equivalent in SQLDF)

employees <- structure(list(id = 1:20, lastname = structure(c(5L, 14L, 13L, 15L, 6L, 16L, 9L, 1L, 3L, 12L, 10L, 8L, 12L, 3L, 11L, 13L, 10L, 7L, 2L, 4L), .Label = c("a", "b", "c", "f", "g", "h", "i", "j", "n", "o", "p", "r", "s", "t", "w", "z"), class = "factor"), firstname = structure(c(12L, 6L, 5L, 12L, 11L, 15L, 9L, 18L, 17L, 7L, 8L, 10L, 4L, 14L, 19L, 16L, 1L, 13L, 2L, 3L), .Label = c("chris", "dima", "drew", "eric", "hila", "jason", "jeremy", "joe", "jon", "jowanza", "lashanda", "matt", "michael", "michelle", "randy", "rudi", "solon", "stewart", "tim"), class = "factor"), gender = structure(c(2L, 2L, 1L, 2L, 1L, 2L, 2L, 2L, 1L, 2L, 2L, 2L, 2L, 1L, 2L, 2L, 2L, 2L, 2L, 2L), .Label = c("f", "m"), class = "factor")), .Names = c("id", "lastname", "firstname", "gender"), class = "data.frame", row.names = c(NA, -20L))

orders <- structure(list(id = c(1L, 1L, 1L, 1L, 1L, 1L, 1L, 2L, 2L, 2L, 2L, 2L, 4L, 4L, 5L, 5L, 5L, NA, NA), item = structure(c(8L, 9L, 12L, 7L, 10L, 13L, 5L, 11L, 8L, 15L, 13L, 16L, 1L, 3L, 14L, 6L, 15L, 2L, 4L), .Label = c("california roll", "chicken teriaki hibachi", "cucumber roll", "diet coke", "edamame", "firecracker roll", "keystone roll", "playboy roll", "rockstar roll", "salmon sashimi", "salmon skin roll", "spider roll", "tuna sashimi", "unagi roll", "unobtanium roll", "yellowtail hand roll"), class = "factor"), quantity\_ordered = c(1L, 1L, 1L, 1L, 6L, 6L, 1L, 1L, 1L, 1L, 4L, 1L, 1L, 1L, 1L, 1L, 1L, 1L, 1L), item\_cost = c(12, 10, 8, 25, 3, 2.5, 6, 8, 12, 35, 2.5, 7, 4, 3.5, 6.5, 9, 35, 7.95, 1.95)), .Names = c("id", "item", "quantity\_ordered", "item\_cost"), class = "data.frame", row.names = c(NA, -19L))

#4 - Use a case statement to define a new data column of california employees,

# using "lower" to evaluate all names as lowercase to ensure case insensitivity

employees\_cali <- sqldf("SELECT \*,

CASE

WHEN lower(firstname) = 'stewart' THEN 1

WHEN lower(firstname) = 'hila' THEN 1

WHEN lower(firstname) = 'jon' THEN 1

WHEN lower(firstname) = 'solon' THEN 1

ELSE 0

END as cali\_emp

FROM employees

")

employees\_cali

#5 - Sort employees\_cali by cali\_emp descending, first name ascending (ascending is default)

employees\_cali\_sorted <- sqldf("SELECT \*,

CASE

WHEN lower(firstname) = 'stewart' THEN 1

WHEN lower(firstname) = 'hila' THEN 1

WHEN lower(firstname) = 'jon' THEN 1

WHEN lower(firstname) = 'solon' THEN 1

ELSE 0

END as cali\_emp

FROM employees

ORDER BY cali\_emp DESC, firstname

")

employees\_cali\_sorted

library(sqldf)

peopleDf <- data.frame(PersonalID=c("ZP1U3EPU2FKAWI6K5US5LDV50KRI1LN7", "IA26X38HOTOIBHYIRV8CKR5RDS8KNGHV", "LASDU89NRABVJWW779W4JGGAN90IQ5B2"),

FirstName=c("Timmy", "Fela", "Sarah"),

LastName=c("Tesa", "Falla", "Kerrigan"),

DOB=c("2010-01-01", "1999-1-1", "1992-04-01"))

##################################################################

peopleDf1 <- sqldf("SELECT \*,

CASE WHEN DOB > '2000-1-1' THEN 'Yes' ELSE 'No' END As 'Millennial'

FROM peopleDf")

peopleDf

peopleDf1

#Using OR with CASE WHEN

peopleDf2 <- sqldf("SELECT \*,

CASE WHEN DOB > '2000-1-1' OR FirstName = 'Sarah' THEN 'PersonIsCool' ELSE 'NotHip' END As 'Cool?'

FROM peopleDf")

peopleDf2

#Using AND with CASE WHEN

peopleDf3 <- sqldf("SELECT \*,

CASE WHEN FirstName = 'Sarah' AND LastName = 'Kerrigan' THEN 'Yes' ELSE ''

END As 'Queen of Blades'

FROM peopleDf")

peopleDf3

#Using SUM with CASE WHEN

count1 <- sqldf("SELECT

SUM(

CASE WHEN Gender = 'Female' THEN 1 ELSE 0 END

) As 'NumberOfFemales',

SUM(

CASE WHEN Gender = 'Male' THEN 1 ELSE 0 END

) As 'NumberOfMales'

FROM peopleDf")

count1

#Using Multiple CASES

peopleDf4 <- sqldf("SELECT \*, CASE WHEN DOB >= '1980-01-01' AND DOB < '1990-01-01' THEN 'X'

WHEN DOB >= '1990-01-01' AND DOB < '2000-01-01' THEN 'Y'

WHEN DOB >= '2000-01-01' AND DOB < '2010-01-01' THEN 'Millennial'

WHEN DOB >= '2010-01-01' AND DOB < '2020-01-01' THEN 'NotYetDefined'

END As 'Generation'

FROM peopleDf")

peopleDf4

Loops:

FOR Loop

for(i in 1:10)

{

print(i)

}



surveys\_adjusted=read.csv(file.choose())

#Example-1:

for (i in 1:dim(surveys\_adjusted)[1]) {

if (surveys\_adjusted$year[i] == 1984) {

print(surveys\_adjusted$weight[i]\*1.1)

} else {

print("It's not 1984.")

}

}

#Example-2:

for (i in 1:dim(surveys\_adjusted)[1]) {

if (surveys\_adjusted$year[i] == 1984) {

print(surveys\_adjusted$weight[i]\*1.1)

}

}

#Example-3:

for (i in 1:dim(surveys\_adjusted)[1]) {

if (surveys\_adjusted$year[i] == 1984) {

surveys\_adjusted$weight1[i] <- surveys\_adjusted$weight[i]\*1.1

}

}

#Example-4:

for (i in 1:dim(surveys\_adjusted)[1]) {

if (surveys\_adjusted$year[i] == 1984) {

surveys\_adjusted$weight1[i] <- surveys\_adjusted$weight[i]\*1.1

}

else {

surveys\_adjusted$weight1[i] <- 0

}

}

a=subset(surveys\_adjusted,year == 1984)

head(a)

head(surveys\_adjusted)

While Loop

count <- 0

while(count < 10)

{

print(count)

count <- count + 1

}

---------------------------

surveys\_adjusted1=head(surveys\_adjusted,10)

num\_rows <- nrow(surveys\_adjusted1)

num\_rows

i <- 1

while (i <= num\_rows) {

surveys\_adjusted1$weight2[i] <- surveys\_adjusted1$weight[i]\*100

#INCREMENT i TO GET NEXT COLUMN

i <- i + 1

}

Break and Next

for (i in 1:3) {

for (j in 3:1) {

if ((i+j) > 4) {

break } else {

print(paste("i=", i, "j=", j))

}

}

}

---------------------------

for (i in 1:3) {

for (j in 3:1) {

if ((i+j) > 4) {

next

} else {

print(paste("i=", i, "j=", j))

}

}

}

Repeat

x <- 1

repeat {

print(x)

x = x+1

if (x == 6){

break

}

}