```
# Load the necessary package
library(genetics)

# Create a simulated dataset of genotypes
data <- matrix(0, 1000, 10)
for (i in 1:10) {
  data[, i] <- sample(c("A/A", "A/a", "a/A", "a/a"), 1000, replace = TRUE)
}

# Convert to genetics object
data <- makeGenotypes(data)

# Initialize containers
af <- matrix(0, 2, 10)  # Allele frequency matrix
gf <- list()  # Genotype frequency list
het <- c()  # Heterozygosity vector
pic <- c()  # Polymorphism informative content vector
phwe <- c()  # P-values for HWE test

# Calculate allele frequencies, genotype frequencies, heterozygosity, and PIC for each SNP
for (i in 1:ncol(data)) {
  r <- summary(data[, i])
  af[, i] <- r$allele.freq[, 2]  # Allele frequencies
  gf[[i]] <- r$genotype.freq[, 2]  # Genotype frequencies
  het[i] <- r$Hu  # Heterozygosity
  pic[i] <- r$pic  # Polymorphism informative content
```

```
  phwe[i] <- HWE.test(data[, i])$test$p.value  # P-value for Hardy-Weinberg equilibrium test
}

# Plot allele frequencies (overlapping plot)
plot(af[1, ], af[2, ], ylim = c(0, 1), pch = 19, col = rgb(0, 0, 1, 0.5), xlab = "Allele 1 Frequency", ylab = "Allele 2 Frequency")
abline(h = 0.5, col = "blue")

# Histogram of p-values for Hardy-Weinberg equilibrium test
hist(phwe, breaks = 30, main = "Histogram of p-values for HWE Test", xlab = "p-value", col = "lightblue", border = "black")


# Filter out p-values less than 0.05 and plot histogram again
phwe_filtered <- phwe[phwe >= 0.05]
hist(phwe_filtered, breaks = 30, main = "Histogram of p-values (p >= 0.05)", xlab = "p-value", col = "pink", border = "black")
abline(h = 0, col = "red")

# Genotypic frequency overlapping barplot
gf_matrix <- matrix(unlist(gf), 3, 100)
barplot(gf_matrix, beside = TRUE, col = c("red", "blue", "green"), legend = c("AA", "Aa", "aa"), ylim = c(0, 1),
      main = "Genotypic Frequency Barplot", xlab = "SNP Index", ylab = "Frequency")

# Plot heterozygosity (with threshold line at 0.5)
```

```
plot(het, ylim = c(0, 1), type = "h", col = "maroon", lwd = 2, main = "Heterozygosity", xlab = "SNP Index", ylab = "Heterozygosity")

abline(h = 0.5, col = "blue")


# Log-transformed heterozygosity plot

plot(log(het), ylim = c(0, log(1)), type = "h", col = "maroon", lwd = 2, main = "Log-transformed Heterozygosity", xlab = "SNP Index", ylab = "Log(Heterozygosity)")

abline(h = log(0.5), col = "blue")


# Plot polymorphism informative content

plot(pic, ylim = c(0, 1), type = "h", col = "purple", lwd = 2, main = "Polymorphism Informative Content (PIC)", xlab = "SNP Index", ylab = "PIC")

abline(h = 0.5, col = "blue")



## this is not perfect plot

par(mfrow=c(1,1))

barplot(af,

    beside = TRUE,

    col = c(rgb(0.2,0.5,0.9,0.8),  # Allele A

        rgb(0.9,0.3,0.3,0.8)), # Allele a

    border = NA,

    ylim = c(0,1),

    xlab = "SNP Index",

    ylab = "Allele Frequency",

    main = "Allele Frequency Barplot (Side-by-Side)",

    cex.main = 1.4,
```

```
      cex.lab = 1.2)

legend("topright",
      legend = c("Allele A", "Allele a"),
      fill = c(rgb(0.2,0.5,0.9,0.8), rgb(0.9,0.3,0.3,0.8)),
      border = NA,
      cex = 1.1)



barplot(af,
       beside = FALSE,      # stacked
       col = c("#3B82F6AA","#EF4444AA"),  # slightly transparent
       border = "white",
       ylim = c(0,1),
       xlab = "SNP Index",
       ylab = "Allele Frequency (Stacked)",
       main = "Allele Frequency Barplot (Stacked)",
       cex.main = 1.4,
       cex.lab = 1.2)

legend("topright",
      legend = c("Allele A","Allele a"),
      fill = c("#3B82F6AA","#EF4444AA"),
      border = NA,
      cex = 1.1)
```

```
# Real data
library(LDheatmap)
data(CEUData)
View(CEUSNP)
class(CEUSNP)
class(CEUDist)
mis<-colSums((is.na(CEUSNP)))
mis
id<-which(mis==0)
id
##
CEUSNP<-CEUSNP[,id]
CEUDist<-CEUDist[id]
dim(CEUSNP); length(CEUDist)
n<-length(CEUSNP)
af<-matrix(0,2,n);gf<-list()
het<-c(); pic<-c(); phwe<-c()
for (i in 1:ncol(CEUSNP)) {
  r <- summary(CEUSNP[, i])
  af[, i] <- r$allele.freq[, 2]
  gf[[i]] <- r$genotype.freq[, 2]
  het[i] <- r$Hu
  pic[i] <- r$pic
  phwe[i] <- HWE.test(CEUSNP[, i])$test$p.value
}
```

```
het
pic
#Overlapping Barplot for allele frequency
colnames(af) <- colnames(CEUSNP)
rownames(af) <- c("Allele1", "Allele2")

# stacked barplot for allele frequency
barplot(af,
        beside = FALSE,
        col = c("steelblue", "pink"),
        legend.text = TRUE,
        args.legend = list(x = "topright"),
        main = "Allele Frequency (Stacked Barplot)",
        ylab = "Frequency",
        las = 2)
maf <- apply(af, 2, min)
maf
low_maf_snps <- names(maf)[maf < 0.05]
low_maf_snps

plot(af[1,],af[2,],ylim=c(0,1))
abline(h=0.05,col="blue")
unlist(gf[[1]])
class(gf[[1]])
unlist(gf)
```

```
matrix(unlist(gf),3,13)
gf[[13]]
gf1<-matrix(unlist(gf),3,13)
##Genotype Frequency Overlapping Barplot
gf_mat <- do.call(cbind, gf)
colnames(gf_mat) <- colnames(CEUSNP)
rownames(gf_mat) <- names(gf[[1]])

# stacked barplot for genotype frequencies
barplot(gf_mat,
        beside = FALSE,
        col = c("pink", "maroon", "skyblue"),
        legend.text = TRUE,
        args.legend = list(x = "topright"),
        main = "Genotype Frequency (Stacked Barplot)",
        ylab = "Frequency",
        las = 2)

plot(het,ylim=c(0,1));abline(h=0.5,col="maroon")
plot(log(het))
abline(h=log(0.5),col="maroon")
#polymorphism informative content
plot(pic,ylim=c(0,1));abline(h=0.5,col="maroon")
maf <- apply(af, 2, min)

# Keep only SNPs with MAF >= 0.05
```

```
keep_snps <- which(maf >= 0.05)
CEUSNP <- CEUSNP[, keep_snps]
af <- af[, keep_snps]

# Overlapping barplot of allele frequencies
barplot(t(af), beside = FALSE, col = c("skyblue", "salmon"),
        legend.text = c("Allele 1", "Allele 2"),
        main = "Overlapping Allele Frequencies",
        xlab = "SNPs", ylab = "Allele Frequency")

status<-c("Case","Control")
phe<- sample(status,nrow(CEUSNP), replace=TRUE)
phe
table(phe,CEUSNP$rs2283089)
tables_list <- list()
# Loop through each SNP
for(i in 1:ncol(CEUSNP)){
  snp_name <- colnames(CEUSNP)[i]
  tab <- table(phe, CEUSNP[, i])
  tables_list[[snp_name]] <- tab
  cat("SNP:", snp_name, "\n")
  print(tab)
}

##Heterozygosity
het <- c()
```

```
for (i in 1:ncol(CEUSNP)) {
  r <- summary(CEUSNP[, i])
  het[i] <- r$Hu
}
# keep SNPs with heterozygosity >= 0.45
keep_id <- which(het >= 0.45)
keep_id
```