```go
package main

import (
    "crypto/md5"
    "fmt"
    "io"
    "os"

    "github.com/go-redis/redis"
)

var client *redis.Client

func init() {
    if client == nil {
        client = newClient()
    }
}

func newClient() *redis.Client {
    client = redis.NewClient(&redis.Options{
        Addr:     "127.0.0.1:6379",
        Password: "",
        //默认有16个数据库,在redis客户端使用select 2 切换
        DB: 2,
    })
    return client
}

const path = "E:\\201708\\20170807-20170813.md"
const bufferKey = "md5hash"

func main() {
    md5String, err := computeAndSaveMd5Hash(path)
    if err != nil {
```

```
        return
    }
    boolValue := isExistBuffer(md5String)
    fmt.Println(boolValue)
}

//计算文件的hash并将hash存在redis中
func computeAndSaveMd5Hash(path string) (string, error) {
    md5String := getFileHash(path)
    fmt.Println(md5String)
    err := client.SAdd(bufferKey, md5String).Err()
    if err != nil {
        return md5String, err
    }
    return md5String, nil
}

//判断文件hash是否已经缓存
func isExistBuffer(md5String string) bool {
    isExist := false
    boolcmd := client.SIsMember(bufferKey, md5String)
    if boolcmd.Err() != nil {
        return isExist
    }
    isExist = boolcmd.Val()
    return isExist
}

// 计算文件哈希
func getFileHash(path string) string {
    file, err := os.Open(path)
    defer file.Close()
    if err != nil {
        return ""
    }
```

```
    h := md5.New()
    _, err = io.Copy(h, file)
    if err != nil {
        return ""
    }
    return fmt.Sprintf("%x", h.Sum(nil))
}
```