问题：

外部系统定时向 MySQL 中插入数据，每 5 分钟插入一次，每次大约 20 万条。需要把 MySQL 中的数据实时导入到 HDFS 中存储。

思路：

1. 从 MySQL 读取数据到 HDFS

使用 jdbc 访问 MySQL，结果保存在 ResultSet 中。

```
//方法 1：在这里直接调用
while( rs.next() ){
    Integer id = rs.getInt(1);
    String name = rs.getString(2);
    System.out.println(id+"\t"+name);
    bytes[] bytes = (id+"\t"+name) .getBytes();

    BytesArrayInputStream inputStream = new BytesArrayInputStream(bytes);
    IOUtils.copyBytes(inputStream, outputStream, conf, close);
}
//方法 1 的缺点，每一行需要写一次 HDFS，效率非常低。


//方法 2：在这里直接调用
StringBuilder sbuilder = new StringBuilder();
while( rs.next() ){
    Integer id = rs.getInt(1);
    String name = rs.getString(2);
    System.out.println(id+"\t"+name);
    sbuilder.append(id+"\t"+name).append("\r\n");
}
byte[] bytes = sbuilder.toString().getBytes();
BytesArrayInputStream inputStream = new BytesArrayInputStream(bytes);
```

```
IOUtils.copyBytes(inputStream, outputStream, conf, close);
```

//方法 2 的缺点：整个结果集写一次 HDFS。如果 MySQL 中查询出的数据量特别大，那么有可能内存溢出。

```
//方法 3：在这里直接调用
StringBuilder sbuilder = new StringBuilder();
int count =0; //记录 sbuilder 中读取的行数
while( rs.next() ){
    count++;
    Integer id = rs.getInt(1);
    String name = rs.getString(2);
    System.out.println(id+"\t"+name);
    sbuilder.append(id+"\t"+name).append("\r\n");
    if(count>10000){
        //写入 HDFS
       byte[] bytes = sbuilder.toString().getBytes();
       write(bytes);
       count=0;
       sbuilder = new StringBuilder();
    }
}
//最后收个尾
byte[] bytes = sbuilder.toString().getBytes();
write(bytes);

void write(byte[] bytes){
   BytesArrayInputStream inputStream = new BytesArrayInputStream(bytes);
   IOUtils.copyBytes(inputStream, outputStream, conf, close);
}
```

```
//方法 4
File file = new File(“tmp.txt”);
while( rs.next() ){
    count++;
    Integer id = rs.getInt(1);
    String name = rs.getString(2);
    FileUtils.write(file,  id+”\t”+name,  true);  //追加到文件中
}
IOUtils.copyBytes(new FileInputStream(file),  outputStream,  conf,  close);
//方法 4 的缺点，每一行需要写一次磁盘文件，效率非常低。
```

2. **如何判断新增数据**

每一次读取数据插入到 HDFS 后，新的数据又插入到 MySQL 中。如何判断新增加的数据哪？

如果有自增的 id 或者时间戳，可以保存每一次读取的所有记录中最后的 id 或者时间戳。建议保持到 redis 中。因为 redis 负责持久化，读写效率高。

如果没有自增的 id 或者时间戳，可以使用 sql 中的 limit m,n。在这里，m 表示从表中开头跳过 m 行，n 表示一次读取多少行。在这里，每一次读取的记录条数累加到 m 中，保存到 redis 中。下一次查询的时候，m 就是原来已经读取过的记录条数。n 取多少合适？因为不知道新插入的记录有多少条，就假设无限大，设定为 999999999。

3. **打成 jar 包**

略

4. **定时执行**

使用 crontab 定时执行。

设定格式  */3  *  *  *  *  shell 脚本

shell 脚本中写什么？内容大致如下：

```
source    /etc/profile

yarn    jar    xxxxx.jar      xxxMainClass
```

为什么使用 yarn 运行 jar，而不是使用 java -jar？因为 yarn 运行的时候，classpath 中含有 hadoop 的 jar 包，可以避免再次设置。