# 参考答案

## 项目一

### 一、选择题

1. D　　2. D　　3. C　　4. A　　5. D　　6. C

### 二、填空题

1. 数据类型多样化、数据价值密度低
2. 数据传输、数据预处理
3. 结构化数据、非结构化数据
4. 控制节点、爬虫节点、资源库
5. 聚焦网络爬虫、增量式网络爬虫、深层网络爬虫

### 三、简答题

略

### 四、操作题

略

## 项目二

### 一、选择题

1. B　　2. C　　3. D　　4. D　　5. C.　　6. B

### 二、填空题

1. 请求头、请求体
2. 状态行
3. 响应头、文字编码
4. 纯粹的 HTML 格式
5. 程序

### 三、简答题

略

### 四、操作题

略

## 项目三

### 一、选择题

1. D　　2. A　　3. B　　4. B　　5. A　　6. A.

### 二、填空题

1. 函数式、类包装式
2. 代码、状态
3. 提前停止
4. 空、退出
5. OUTSTANDING、PROCESSING、COMPLETE

### 三、简答题

略

### 四、操作题

1. /*. html
2. requests. url

## 项目四

### 一、选择题

1. D　　2. B　　3. D　　4. D　　5. A　　6. B

### 二、填空题

1. 命令行
2. genspider、genspider、start urls
3. start urls、Downloader
4. css
5. DOWNLOAD DELAY

### 三、简答题

略

### 四、操作题

分析如下：在爬虫文件中，构造 xpath 表达式来获取电影的名称并输出到控制台上。使用 xpath 表达式定位电影名称后，再调用 extract（ ）方法提取目标数据，并通过 for 循环输出到控制台上。

关键代码如下所示。

```
from matplotlib. pyplot import title
import scrapy

class MovieSpider(scrapy.Spider):
```

```
        name='itcast'
        allowed domains=['movie.douban.com']
        start_urls=['https://movie.douban.com/top250']
        def parse(self,response):
        titles=response.xpath('//ol[@class="grid_view"]//div[@ class="hd"]/
a/span[1]/text()').extract()
            print(titles)
```

## 项目五

### 一、选择题

1. A　　2. D　　3. D　　4. C　　5. A　　6. D

### 二、填空题

1. Request、Response

2. 通过 IP 与访问间隔等请求模式识别、通过 Header 内容识别、通过 cookies 信息识别、通过验证码识别、通过对客户端是否支持 JS 的分析来识别、通过是否遵循 Robots 协议来识别、通过页面资源是否加载来识别（任意 4 个即对）

3. 行为动作验证、第三方验证

4. 机器图像识别、人工打码

5. 尽可能真实地模拟用户访问

### 三、简答题

略

### 四、操作题

略