



Java 实战练习题：身份证号码工具类 **IdCardUtils**



题目名称：

编写一个中国大陆身份证号码处理工具类 **IdCardUtils**，用于对身份证进行校验、信息提取和脱敏处理。



背景说明：

在中国大陆，公民身份证号码具有严格的结构规则，通常为 18 位数字加校验码。身份证号中包含了出生日期、性别、省市区代码等信息。本题旨在引导你熟悉字符串处理、正则表达式、日期处理、数字运算等核心 Java 编程技巧。



目标功能：



1. **public static String maskIdCard(String id)**

- 功能：身份证脱敏（用于保护隐私），将生日部分用星号代替。
- 实现要求：
 - 前6位 + ***** + 后4位
 - 示例：110105199001012345 → 110105*****2345



2. **public static String getGender(String id)**

- 功能：根据身份证第17位判断性别（奇数为男，偶数为女）
- 示例：
 - 330825199610281021 → 性别码是第17位 为2 → 女
 - 34120219890305363x → 性别码是 3 → 男



3. **public static int getAge(String id)**

- 功能：从身份证中提取出生年月，并计算到当前日期的年龄（整岁）。

- 实现要求：
 - 出生日期位置为第 7~14 位 (yyyyMMdd)
 - 使用 `LocalDate` 和 `Period` 进行年龄计算

4. `public static boolean isValidIdCard(String id)`

- 功能：判断输入身份证号是否合法。
- 实现要求：
 - 长度必须是 18 位
 - 前17位为数字，第18位为数字或大小写 X
 - 出生年月日合法，且格式为 yyyyMMdd，支持的范围：1800年-2399年
 - 校验码必须正确（使用国家标准 GB 11643-1999 的校验算法，详见附录）
- 合法身份证号参考格式示例：
 - 330825199610281021
 - 34120219890305363x
 - 341225199007080823
 - 320826198801393422

📎 附录：身份证号第18位校验码算法说明（国家标准 GB 11643-1999）

中国大陆身份证号码的最后一位（第18位）是一个校验码，根据前17位数字计算得出，用于防止伪造。

✅ 校验码用途

用于检测身份证号是否被篡改，是数据安全的一种校验机制。

🧠 计算规则（共4步）

1 2 3 4 步骤 1：准备前17位数字

身份证号前17位是纯数字，用于后续计算。

示例号码前17位：

身份证号：330825199610281021

下标	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
提取	3	3	0	8	2	5	1	9	9	6	1	0	2	8	1	0	2

🏠 步骤 2：准备对应加权因子数组

身份证号的前17位，每一位对应一个加权因子：

下标	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
因子	7	9	10	5	8	4	2	1	6	3	7	9	10	5	8	4	2

✚ 步骤 3：逐位相乘并累加求和

身份证每位数字 × 对应的权重，然后求和：

$\text{sum} = \text{id}[0]*7 + \text{id}[1]*9 + \text{id}[2]*10 + \dots + \text{id}[16]*2$

以示例 330825199610281021 为例：

$3*7 + 3*9 + 0*10 + 8*5 + 2*8 + 5*4 + 1*2 + 9*1 + 9*6 + 6*3 +$
 $1*7 + 0*9 + 2*10 + 8*5 + 1*8 + 0*4 + 2*2$
 $= 286$

步骤 4：根据求和结果取模，映射校验码

将上一步的 $\text{sum} \% 11$ ，得到一个 0–10 之间的整数。

然后查表得到校验码：

余数 (mod 11)	校验码
0	1
1	0
2	X
3	9
4	8
5	7
6	6
7	5
8	4
9	3
10	2

比较计算得出的校验码是否与身份证第18位一致（忽略大小写）

例如：

- 如果结果是 0 → 校验码应为 1
- 如果身份证第18位是 x（小写），也是合法的