Which One is Better?

- **For Simple Logic (Single Condition)**: If you're dealing with a single condition (e.g., checking whether `@A` is less than 10), the `IF` statement with `SET` might be more straightforward and easier to read.

```sql
IF @A < 10 SET @C = CONCAT('C00', @A);
```

- **For Multiple Conditions**: If you need to handle multiple conditions and format the value based on those conditions (e.g., formatting `@I` with different numbers of leading zeros), the `CASE` expression is definitely more efficient. You can handle all the conditions in a single statement without needing multiple `IF` statements.

```sql
SELECT @R = CASE WHEN @I < 10 THEN CONCAT(@C, '000', @I) WHEN @I < 100 THEN CONCAT(@C, '00', @I) WHEN @I < 1000 THEN CONCAT(@C, '0', @I) ELSE CONCAT(@C, @I) END
```

Conclusion:

- **Use `IF` and `SET`** when you have simple, straightforward conditions with minimal formatting.
- **Use `CASE`** when you need to handle multiple conditions in one go, especially if you're formatting the value based on those conditions (e.g., leading zeros). It provides a more compact and scalable solution.

In your scenario, if you want flexibility and scalability, **the `CASE` expression is generally the better choice** for auto-generating IDs with varying formats.