### **Name**

| **Label** | **Description** | **Generated String** | **Example Output** |
| --- | --- | --- | --- |
| **N** | **Full name without spaces** | tmp[1] **-** ZhangSanChuan | **"ZhangSanChuan"** |
| **a** | **Secondary identifier followed by the first name** | tmp[2] **+** name\_str[0] **-** SanChuanZhang | **"SanChuanZhang"** |
| **b** | **Short form identifier (initials)** | tmp[0] **-** ZSC | **"ZSC"** |
| **c** | **Secondary identifier** | tmp[2] **-** SanChuan | **"SanChuan"** |
| **d** | **First word only** | name\_str[0] **-** Zhang | **"Zhang"** |
| **e** | **First word with the first letter capitalized** | tmp[3] **-** Zhang | **"Zhang"** |
| **f** | **Initials after the first, followed by first word** | tmp[4] **+** name\_str[0] **-** SCZhang | **"SCZhang"** |
| **g** | **First word followed by the initials after the first** | name\_str[0] **+** tmp[4] **-** ZhangSC | **"ZhangSC"** |

### 

### 

### **Birth Processing Rules**

| **Label** | **Condition** | **Description** | **Generated String** | **Example Output** |
| --- | --- | --- | --- | --- |
| B | None | Full birthdate | birth | "19940231" |
| h | Month starts with '0' | Year followed by day, omitting zero in month | birth.substr(0, 4) + birth.substr(5, 3) | "1994231" |
| i | Day tens digit is '0' | Full birthdate omitting zero in day tens place | birth.substr(0, 6) + birth[7] | "1994021" |
| j | Month and day tens digit are '0' | Year followed by month unit and day unit | birth.substr(0, 4) + birth[5] + birth[7] | "199421" |
| k | None | Year and month of birthdate | birth.substr(0, 6) | "199402" |
| l | None | Last six digits of birthdate | birth.substr(2, 6) | "940231" |
| m | None | Year of birthdate | birth.substr(0, 4) | "1994" |
| n | None | Month and day of birthdate | birth.substr(4, 4) | "0231" |
| o | None | Last four digits starting from year | birth.substr(2, 4) | "9402" |
| p | None | Day and month followed by year | birth.substr(4, 4) + birth.substr(0, 4) | "02311994" |
| q | None | Month followed by year | birth.substr(4, 2) + birth.substr(0, 4) | "021994" |
| r | None | Month and day followed by last two digits of year | birth.substr(4, 4) + birth.substr(2, 2) | "023194" |

### **Email Processing Rules**

| **Label** | **Condition** | **Description** | **Generated String** | **Example Output** |
| --- | --- | --- | --- | --- |
| E | None | Email prefix before the '@' symbol | email.substr(0, pos) | "example" |
| s | Prefix contains digits | Extracts consecutive digits from the start of the prefix | email\_d | "123" (from "123abc@example.com") |
| t | Prefix contains letters | Extracts consecutive letters from the start of the prefix | email\_l | "abc" (from "abc123@example.com") |

### **Account Processing Rules**

| **Label** | **Condition** | **Description** | **Generated String** | **Example Output** |
| --- | --- | --- | --- | --- |
| A | None | Processes the entire account string | account | "user123" |
| u | Account starts with digits | Extracts consecutive digits from the start of the account | acc\_d | "123" (from "123user") |
| v | Account starts with letters | Extracts consecutive letters from the start of the account | acc\_l | "user" (from "user123") |

### **GID Processing Rules**

| **Label** | **Condition** | **Description** | **Generated String** | **Example Output** |
| --- | --- | --- | --- | --- |
| G | None | Processes the entire gid string | gid | "ABCDEFGH0123456789" |
| w | None | Extracts the last 4 characters of the gid | gid.substr(len - 4, 4) | "6789" |

### **Phone Processing Rules**

| **Label** | **Condition** | **Description** | **Generated String** | **Example Output** |
| --- | --- | --- | --- | --- |
| C | None | Processes the entire phone number string | phone | "123-456-7890" |