

Chapter 1

Testing

1.1 Define statement

The first thing to do is give a name to the new tagset. That is done with the define statement. Our tagset is going to be called `plain_text`. The tagset should be put in `proc template`'s tagset directory, so its first name is `tagsets.`. The code in Figure 1.1 shows a very simple tagset definition.

Code 1.1: *Defining a Tagset*

```
Define tagset tagsets.plain_text;

end;
```

1.1.1 A listing example

This is a very simple example of listing. The code shown if Figure 1.1 was created using the listing package. This example is a direct usage of the listing environment, I've added arguments to make it look nearly the same as the `fancyvrb` example above. The next example 1.2 on the following page uses the `lst` environment created above. Notice the `varioref` for 1.2 adapts nicely to the location of the listing. The environment makes it easier to use and allows for style changes in one spot.

Listing 1.1: SSV listing

```
proc template;

  Define tagset tagsets.SSV;

    parent=tagsets.csv;

    define event header;
      start:
        put ';' / if !cmp(COLSTART, "1");
        put '"';
        put VALUE;
    end;
  end;
end;
```

```

        finish:
            put '""';
    end;

    define event data;
        start:
            put ';' / if !cmp(COLSTART, "1");
            put '""';
            put VALUE;
        finish:
            put '""';
    end;

    define event colspanfill;
        put ';';
    end;

    define event rowspanfill;
        put ';' /if ! exists(VALUE);
    end;

end;

run;

options obs=3;
ods tagsets.ssv file='test.ssv';
proc print data=sashelp.class;run;
ods tagsets.ssv close;

```

Listing 1.2: SSV listing2

```

proc template;

    Define tagset tagsets.SSV;

        parent=tagsets.csv;

        define event header;
            start:
                put ';' / if !cmp(COLSTART, "1");
                put '""';
                put VALUE;
            finish:
                put '""';
        end;

        define event data;

```

```
        start:
            put ';' / if !cmp(COLSTART, "1");
            put '"';
            put VALUE;
        finish:
            put '"';
    end;

    define event colspanfill;
        put ' ';
    end;

    define event rowspanfill;
        put ';' /if ! exists(VALUE);
    end;

end;

run;

options obs=3;
ods tagsets.ssv file='test.ssv';
proc print data=sashelp.class;run;
ods tagsets.ssv close;
```
