Equivalent \( \Sigma, F \) derivations, Phrase Markers, and Reduced Phrase Markers

\( \Sigma: S \)

\( F: \)

\[
S \rightarrow NP \ VP \quad VP \rightarrow V \ NP \\
NP \rightarrow John \quad V \rightarrow likes \\
NP \rightarrow Mary
\]

The equivalent \( \Sigma, F \) derivations of "John likes Mary", given the above \( \Sigma, F \) grammar:

\[
\begin{array}{llll}
S & S & S \\
NP & VP & NP & VP & NP & VP \\
John & VP & John & VP & John & VP \\
John & V & NP & John & V & NP & John & V & NP \\
John & likes & NP & John & V & Mary & John & likes & NP \\
John & likes & Mary & John & likes & Mary & John & likes & Mary \\
\end{array}
\]

The Phrase Marker of "John likes Mary", given the above \( \Sigma, F \) grammar:

\{S, NP VP, NP V NP, NP likes Mary, NP V Mary, NP likes NP, John VP, John V NP, John V Mary, John likes NP, John likes Mary\}

[This set consists of all the lines occurring in any of the equivalent derivations of "John likes Mary", given the above \( \Sigma, F \) grammar]

The Reduced Phrase Marker of "John likes Mary", given the above \( \Sigma, F \) grammar:

\{S, NP likes Mary, John VP, John V Mary, John likes NP, John likes Mary\}

[This set is the subset of the Phrase Marker consisting of the terminal string plus all the monostrings (the strings containing exactly one non-terminal surrounded by any number of terminal symbols) Lasnik and Kupin observed that this set suffices to compute all the 'is a' relations.]

Collapsed derivation tree for "John likes Mary", given the above \( \Sigma, F \) grammar:

```
S
 | 
NPVP
 | 
JohnVPNP
 | 
likesMary
```