```
void Cell::Set(const std::string& text) {
if (text.size() > 1 && text[0] == FORMULA_SIGN) {
std::unique_ptr<Impl> impl;
impl = std::make_unique<FormulaImpl>(std::move(text.substr(1)), sheet_);
FindCyclicDependencies(impl->GetReferencedCells());
impl_ = std::move(impl);
else if (text.empty()) {
impl = std::make unique<EmptyImpl>();
impl_ = std::make_unique<TextImpl>(text);
for (auto cell : dependent_cells_) {
dynamic_cast<Cell*>(cell)->cache_cells_.erase(this);
dependent_cells_.clear();
for (auto& cell : impl_->GetReferencedCells()) {
auto c = sheet_.GetCell(cell);
if (c == nullptr) {
sheet_.SetCell(cell, {});
c = sheet .GetCell(cell);
dependent cells .insert(c);
dynamic_cast<Cell*>(c)->SetDependentCell(this);
UpdateCache();
```

