Hourly Visualisation and Panel Compilation

#Visualisation Model v Observed Hourly  
#Thomas Zwagerman  
#Sat 10th of March  
####  
library(tidyr)  
library(dplyr)

library(ggplot2)  
library(readr)  
library(gridExtra)

library(colourpicker)  
library(forecast)

Desktop/toolik\_timeseries/toolik\_timeseries")  
  
#Reading in csv's----  
observed <- read.csv('observed\_hourly.csv', header = TRUE, skip = 0, sep = ",")  
soiltemp <- read.csv('default\_3hourly.csv', header = TRUE, skip = 0, sep = ",")  
modeldefault <- read.csv('default\_hourly.csv', header = TRUE, skip = 0, sep = ",")  
green <- read.csv('green\_hourly.csv', header = TRUE, skip = 0, sep = ",")  
evergreen <- read.csv('evergreen\_hourly.csv', header = TRUE, skip = 0, sep = ",")  
real <- read.csv('realistic\_hourly.csv', header = TRUE, skip = 0, sep = ",")  
differences <- read.csv('1-differences\_hourly.csv', header = TRUE, skip = 0, sep = ",")  
  
#Changing factor to date class----  
observed$date\_time <- as.POSIXct(observed$date\_time, format = "%Y-%m-%d %H:%M:%S")  
soiltemp$date\_time <- as.POSIXct(soiltemp$date\_time, format = "%Y-%m-%d %H:%M:%S")  
modeldefault$ts\_hour <- as.POSIXct(modeldefault$ts\_hour, format = "%Y-%m-%d %H:%M:%S")  
green$ts\_hour <- as.POSIXct(green$ts\_hour, format = "%Y-%m-%d %H:%M:%S")  
evergreen$ts\_hour <- as.POSIXct(evergreen$ts\_hour, format = "%Y-%m-%d %H:%M:%S")  
real$ts\_hour <- as.POSIXct(real$ts\_hour, format = "%Y-%m-%d %H:%M:%S")  
differences$ts\_hour <- as.POSIXct(differences$ts\_hour, format = "%Y-%m-%d %H:%M:%S")  
  
#Making own theme----  
#Comparison theme  
theme\_toolik <- function(){  
 theme\_classic()+  
 theme(axis.text.x=element\_text(size=12, angle=45, vjust=1, hjust=1), #changing font size, and text at an angle  
 axis.text.y=element\_text(size=12),  
 axis.title.x=element\_text(size=14, face="plain"), #changing font size of axis titles  
 axis.title.y=element\_text(size=14, face="plain"), #plain changes font type, could be italic  
 panel.grid.major.x =element\_blank(), #removed grey grid lines  
 panel.grid.minor.x =element\_blank(),  
 panel.grid.major.y =element\_blank(),  
 panel.grid.minor.y =element\_blank(),  
 plot.margin = unit(c(1,1,1,1), units= , "cm")) + #adding 1cm margin  
 theme(legend.text = element\_text(size=12, face="italic"), #setting font for a legend  
 legend.title = element\_blank(), #removing legend title, blank  
 legend.position=c(0.3, 0.8)) #setting position for legend, 0 is bottom left, 1 is top right.   
}  
  
#Difference theme  
theme\_difference <- function(){  
 theme\_classic()+  
 theme(axis.text.x=element\_text(size=12, angle=45, vjust=1, hjust=1), #changing font size, and text at an angle  
 axis.text.y=element\_text(size=12),  
 axis.title.x=element\_text(size=14, face="plain"), #changing font size of axis titles  
 axis.title.y=element\_text(size=14, face="plain"), #plain changes font type, could be italic  
 panel.grid.major.x =element\_blank(), #removed grey grid lines  
 panel.grid.minor.x =element\_blank(),  
 panel.grid.major.y =element\_blank(),  
 panel.grid.minor.y =element\_blank(),  
 plot.margin = unit(c(1,1,1,1), units= , "cm")) + #adding 1cm margin  
 theme(legend.text = element\_text(size=9, face="italic"), #setting font for a legend  
 legend.title = element\_blank(), #removing legend title, blank  
 legend.position= c(0.4, 0.9)) #setting position for legend, 0 is bottom left, 1 is top right.   
}  
  
theme\_nolegend <- function(){  
 theme\_classic()+  
 theme(axis.text.x=element\_text(size=12, angle=45, vjust=1, hjust=1), #changing font size, and text at an angle  
 axis.text.y=element\_text(size=12),  
 axis.title.x=element\_text(size=14, face="plain"), #changing font size of axis titles  
 axis.title.y=element\_text(size=14, face="plain"), #plain changes font type, could be italic  
 panel.grid.major.x =element\_blank(), #removed grey grid lines  
 panel.grid.minor.x =element\_blank(),  
 panel.grid.major.y =element\_blank(),  
 panel.grid.minor.y =element\_blank(),  
 plot.margin = unit(c(1,1,1,1), units= , "cm")) + #adding 1cm margin  
 theme(legend.position= "none") #setting position for legend, 0 is bottom left, 1 is top right.   
}  
  
#Visualisation output----  
#Air temperature----  
air\_temp <- (ggplot() +  
 geom\_line(data = modeldefault, aes(x = ts\_hour, y = t, colour = "a")) +  
 geom\_line(data = green, aes(x = ts\_hour, y = t, colour = "b")) +  
 geom\_line(data = evergreen, aes(x = ts\_hour, y = t, colour = "c")) +  
 geom\_line(data = real, aes(x = ts\_hour, y = t, colour = "d")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_toolik()+  
 scale\_fill\_manual(values = c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"),  
 labels=c("Default", "Green", "Evergreen", "Real"))+ #adding legend labels  
 ylab("Air Temperature "\*" in"~degree\*C)+  
 xlab("Date (in 2015)"))  
  
#Soil surface temp----  
surface\_temp <- (ggplot() +  
 geom\_line(data = modeldefault, aes(x = ts\_hour, y = tslb.1., colour = "a")) +  
 geom\_line(data = green, aes(x = ts\_hour, y = tslb.1., colour = "b")) +  
 geom\_line(data = evergreen, aes(x = ts\_hour, y = tslb.1., colour = "c")) +  
 geom\_line(data = real, aes(x = ts\_hour, y = tslb.1., colour = "d")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_toolik()+  
 scale\_fill\_manual(values = c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"),  
 labels=c("Default", "Green", "Evergreen", "Real"))+ #adding legend labels  
 ylab("Soil Surface Temperature "\*" in"~degree\*C)+  
 xlab("Date (in 2015)"))  
  
#Soil heat flux ----  
surface\_flux <- (ggplot() +  
 geom\_line(data = modeldefault, aes(x = ts\_hour, y = grdflx, colour = "a")) +  
 geom\_line(data = green, aes(x = ts\_hour, y = grdflx, colour = "b")) +  
 geom\_line(data = evergreen, aes(x = ts\_hour, y = grdflx, colour = "c")) +  
 geom\_line(data = real, aes(x = ts\_hour, y = grdflx, colour = "d")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_nolegend()+  
 scale\_fill\_manual(values = c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"),  
 labels=c("Default", "Green", "Evergreen", "Real"))+ #adding legend labels  
 ylab(bquote('Radiation in' ~W/M^-2))+  
 xlab("Date (in 2015)"))  
  
#Surface Sensible Heat Flux----  
sens\_heat <- (ggplot() +  
 geom\_line(data = modeldefault, aes(x = ts\_hour, y = hfx, colour = "a")) +  
 geom\_line(data = green, aes(x = ts\_hour, y = hfx, colour = "b")) +  
 geom\_line(data = evergreen, aes(x = ts\_hour, y = hfx, colour = "c")) +  
 geom\_line(data = real, aes(x = ts\_hour, y = hfx, colour = "d")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_toolik()+  
 scale\_fill\_manual(values = c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"),  
 labels=c("Default", "Green", "Evergreen", "Real"))+ #adding legend labels  
 ylab(bquote('Radiation in' ~W/M^-2))+  
 xlab("Date (in 2015)"))  
  
#Surface Latent Heat Flux----  
lat\_heat <- (ggplot() +  
 geom\_line(data = modeldefault, aes(x = ts\_hour, y = lh, colour = "a")) +  
 geom\_line(data = green, aes(x = ts\_hour, y = lh, colour = "b")) +  
 geom\_line(data = evergreen, aes(x = ts\_hour, y = lh, colour = "c")) +  
 geom\_line(data = real, aes(x = ts\_hour, y = lh, colour = "d")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_nolegend()+  
 scale\_fill\_manual(values = c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"),  
 labels=c("Default", "Green", "Evergreen", "Real"))+ #adding legend labels  
 ylab(bquote('Radiation in' ~W/M^-2))+  
 xlab("Date (in 2015)"))  
  
#Downward Longwave Radiation----  
long\_rad <- (ggplot() +  
 geom\_line(data = modeldefault, aes(x = ts\_hour, y = glw, colour = "a")) +  
 geom\_line(data = green, aes(x = ts\_hour, y = glw, colour = "b")) +  
 geom\_line(data = evergreen, aes(x = ts\_hour, y = glw, colour = "c")) +  
 geom\_line(data = real, aes(x = ts\_hour, y = glw, colour = "d")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_nolegend()+  
 scale\_fill\_manual(values = c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"),labels=c("Default", "Green", "Evergreen", "Real"))+ #adding legend labels  
 ylab('Radiation in' ~W/M^-2)+  
 xlab("Date (in 2015)"))  
  
#Net Shortwave radiation flux at the ground----  
short\_rad <- (ggplot() +  
 geom\_line(data = modeldefault, aes(x = ts\_hour, y = gsw, colour = "a")) +  
 geom\_line(data = green, aes(x = ts\_hour, y = gsw, colour = "b")) +  
 geom\_line(data = evergreen, aes(x = ts\_hour, y = gsw, colour = "c")) +  
 geom\_line(data = real, aes(x = ts\_hour, y = gsw, colour = "d")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_toolik()+  
 scale\_fill\_manual(values = c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"),  
 labels=c("Default", "Green", "Evergreen", "Real"))+ #adding legend labels  
 ylab('Radiation in' ~W/M^-2)+  
 xlab("Date (in 2015)"))  
  
#Skin temperature----  
skin\_temp <- (ggplot() +  
 geom\_line(data = modeldefault, aes(x = ts\_hour, y = tsk, colour = "a")) +  
 geom\_line(data = green, aes(x = ts\_hour, y = tsk, colour = "b")) +  
 geom\_line(data = evergreen, aes(x = ts\_hour, y = tsk, colour = "c")) +  
 geom\_line(data = real, aes(x = ts\_hour, y = tsk, colour = "d")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_nolegend()+  
 scale\_fill\_manual(values = c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#EE2C2C","b" = "#66CD00", "c" = "#006400","d"= "#FF7F00"),  
 labels=c("Default", "Green", "Evergreen", "Real"))+ #adding legend labels  
 ylab("Skin Temperature"\*"in"~degree\*C)+  
 xlab("Date (in 2015)"))  
  
#Plotting differences----  
#Air temperature----  
air\_temp\_diff <- (ggplot() +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_gr\_t, colour = "a")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_evgr\_t, colour = "b")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_real\_t, colour = "c")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_difference()+  
 scale\_fill\_manual(values = c("a" = "#66CD00","b"="#006400","c"="#F77F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#66CD00","b"= "#006400","c"= "#F77F00"),  
 labels=c("Difference between Green and Default",   
 "Difference between Evergreen and Default",  
 "Difference between Real and Default"))+ #adding legend labels  
 ylab("Air Temperature "\*"in"~degree\*C)+  
 xlab("Date (in 2015)"))  
  
#Differences from default in soil surface temperature----  
surface\_temp\_diff <- (ggplot() +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_gr\_tslb.1., colour = "a")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_evgr\_tslb.1., colour = "b")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_real\_tslb.1., colour = "c")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_difference()+  
 scale\_fill\_manual(values = c("a" = "#66CD00","b"="#006400","c"="#F77F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#66CD00","b"= "#006400","c"= "#F77F00"),  
 labels=c("Difference between Green and Default",   
 "Difference between Evergreen and Default",  
 "Difference between Real and Default"))+ #adding legend labels  
 ylab("Soil Surface Temperature "\*"in"~degree\*C)+  
 xlab("Date (in 2015)"))  
  
#Difference from default in soil heat flux temp----  
surface\_flux\_diff <- (ggplot() +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_gr\_grdflx, colour = "a")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_evgr\_grdflx, colour = "b")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_real\_grdflx, colour = "c")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_nolegend()+  
 scale\_fill\_manual(values = c("a" = "#66CD00","b"="#006400","c"="#F77F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#66CD00","b"= "#006400","c"= "#F77F00"),  
 labels=c("Difference between Green and Default",   
 "Difference between Evergreen and Default",  
 "Difference between Real and Default"))+ #adding legend labels  
 ylab(bquote('Radiation in' ~W/M^-2))+  
 xlab("Date (in 2015)"))  
  
#Differences from default in surface sensible heat flux----  
sens\_heat\_diff <- (ggplot() +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_gr\_hfx, colour = "a")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_evgr\_hfx, colour = "b")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_real\_hfx, colour = "c")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_nolegend()+  
 scale\_fill\_manual(values = c("a" = "#66CD00","b"="#006400","c"="#F77F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#66CD00","b"= "#006400","c"= "#F77F00"),  
 labels=c("Difference between Green and Default",   
 "Difference between Evergreen and Default",  
 "Difference between Real and Default"))+ #adding legend labels  
 ylab(bquote('Radiation in' ~W/M^-2))+  
 xlab("Date (in 2015)"))  
  
#Differences from default in surface latent heat flux----  
lat\_heat\_diff <- (ggplot() +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_gr\_lh, colour = "a")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_evgr\_lh, colour = "b")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_real\_lh, colour = "c")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_difference()+  
 scale\_fill\_manual(values = c("a" = "#66CD00","b"="#006400","c"="#F77F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#66CD00","b"= "#006400","c"= "#F77F00"),  
 labels=c("Difference between Green and Default",   
 "Difference between Evergreen and Default",  
 "Difference between Real and Default"))+ #adding legend labels  
 ylab(bquote('Radiation in' ~W/M^-2))+  
 xlab("Date (in 2015)"))  
  
#Differences from default in downward longwave radiation----  
long\_rad\_diff <- (ggplot() +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_gr\_glw, colour = "a")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_evgr\_glw, colour = "b")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_real\_glw, colour = "c")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_nolegend()+  
 scale\_fill\_manual(values = c("a" = "#66CD00","b"="#006400","c"="#F77F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#66CD00","b"= "#006400","c"= "#F77F00"),  
 labels=c("As above",   
 "As above",  
 "As above"))+ #adding legend labels  
 ylab('Radiation in' ~W/M^-2)+  
 xlab("Date (in 2015)"))  
  
#Differences from default in net shortwave radiation flux at the ground----  
short\_rad\_diff <- (ggplot() +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_gr\_gsw, colour = "a")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_evgr\_gsw, colour = "b")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_real\_gsw, colour = "c")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_difference()+  
 scale\_fill\_manual(values = c("a" = "#66CD00","b"="#006400","c"="#F77F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#66CD00","b"= "#006400","c"= "#F77F00"),  
 labels=c("Difference between Green and Default",   
 "Difference between Evergreen and Default",  
 "Difference between Real and Default"))+ #adding legend labels  
 ylab('Radiation in' ~W/M^-2)+  
 xlab("Date (in 2015)"))  
  
#Differences from default in skin temperature----  
skin\_temp\_diff <- (ggplot() +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_gr\_tsk, colour = "a")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_evgr\_tsk, colour = "b")) +  
 geom\_line(data = differences, aes(x = ts\_hour, y = diff\_real\_tsk, colour = "c")) +  
 scale\_x\_datetime(date\_labels = "%D", date\_breaks = "3 day")+  
 theme\_nolegend()+  
 scale\_fill\_manual(values = c("a" = "#66CD00","b"="#006400","c"="#F77F00"))+ #custom colours  
 scale\_colour\_manual(values=c("a" = "#66CD00","b"= "#006400","c"= "#F77F00"),  
 labels=c("Difference between Green and Default",   
 "Difference between Evergreen and Default",  
 "Difference between Real and Default"))+ #adding legend labels  
 ylab("Skin Temperature "\*"in"~degree\*C)+  
 xlab("Date (in 2015)"))  
  
#Arranging into Panels ----  
panel <- grid.arrange(air\_temp + ggtitle("(a) Air Temperature") +   
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 air\_temp\_diff + ggtitle("(b) Air Temperature Differences") +  
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 skin\_temp + ggtitle("(c) Skin Temperature") +   
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 skin\_temp\_diff + ggtitle("(d) Skin Temperature Differences") +  
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")))

panel\_surface <-grid.arrange(surface\_temp + ggtitle("(a) Surface Temperature") +   
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 surface\_temp\_diff + ggtitle("(b) Surface Temperature Differences") +  
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 surface\_flux + ggtitle("(c) Soil Heat Flux") +  
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 surface\_flux\_diff + ggtitle("(d) Soil Heat Flux Differences") +  
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")))

panel\_senslat <- grid.arrange(sens\_heat + ggtitle("(a) Surface Sensible Heat Flux") +   
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 sens\_heat\_diff + ggtitle("(b) Surface Sensible Heat Flux Differences") +  
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 lat\_heat + ggtitle("(c) Surface Latent Heat Flux") +   
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 lat\_heat\_diff + ggtitle("(d) Surface Latent Heat Flux Differences") +  
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")))

panel\_shortlong <- grid.arrange(short\_rad + ggtitle("(a) Shortwave Radiation") +   
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 short\_rad\_diff + ggtitle("(b) Shortwave Radiation Differences") +  
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 long\_rad + ggtitle("(c) Longwave Radiation") +   
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")),  
 long\_rad\_diff + ggtitle("(d) Longwave Radiation Differences") +  
 theme(plot.margin = unit(c(0.2, 0.2, 0.2, 0.2), units = "cm")))

#Saving panels----  
ggsave(panel, file = "air\_temp\_panel.png", width = 10, height =8)  
ggsave(panel\_surface, file = "surface\_temp\_panel.png", width = 10, height = 8)  
ggsave(panel\_senslat, file = "senslat\_flux\_panel.png", width = 10, height = 8)  
ggsave(panel\_shortlong, file = "shortlong\_rad\_panel.png", width = 10, height = 8)