
******* Methods used in <expression> block *******

******* AI *******

METHOD:

AI.IsSeaZonePatrolled (int countryID, int SeaZone, int PatrolType)

DESCRIPTION:

Returns 1 if value is correct.

Which means that country is patrolling sea zone.

0 - Regular fleet

1 - Raiders

******* CITY *******

METHOD:

City.WasBombed (x, y)

DESCRIPTION:

Checks if the city is still affected by strategic bombardment.

******* COMMANDER *******

METHOD:

Commander.GetLevel (int CommanderID)

DESCRIPTION:

Returns level of commander with given ID

******* COUNTRY GET *******

METHOD:

Country.GetAiCarrierTargetsPriorities (int CountryID, int CarrierPriorityID)

DESCRIPTION:

Returns requested priority for given country.

METHOD:

Country.GetAIP (int countryID, int AIP)

DESCRIPTION:

Returns 1 if value is correct.

METHOD:

Country.GetAlliance (int CountryID)

DESCRIPTION:

Returns Alliance of country with given ID.

Possible returned values are:

- 0 - neutral country,
 - 1 - member of Allies,
 - 2 - member of Axis,
 - 3 - member of Comintern,
 - 4 - leader of Allies,
 - 5 - leader of Axis,
 - 6 - leader of Comintern
-

METHOD:

Country.GetAllianceEntry (int CountryID)

DESCRIPTION:

Returns Alliance Entry of given country

METHOD:

Country.GetDesiredLandForcesStructure (int countryID, int unitType, intWeight)

DESCRIPTION:

Checks desired land forces structure for given unit type for given country.

METHOD:

Country.GetDifficultyLevel (countryID)

DESCRIPTION:

Checks what is difficulty of given country

0-Very Easy

1-Easy

2-Normal

3-Hard

4-Very Hard

METHOD:

Country.GetDiplomaticPoints (int CountryID)

DESCRIPTION:

Returns Diplomatic Points of given country

METHOD:

Country.GetMaxCoveringUnitDistance (int countryID, int value)

DESCRIPTION:

Checks max covering unit distance of the AI for given country.

METHOD:

Country.GetMaxNumberOfFrontsForSealInvasion (int countryID, int value)

DESCRIPTION:

Returns 1 if value is correct.

METHOD:

Country.GetMaxSafeMoveCost (int countryID, int value)

DESCRIPTION:

Checks max safe movement cost of the AI for given country.

METHOD:

Country.GetMaxUnitCountOnFront (int countryID, int value)

DESCRIPTION:

Checks max number of units at front of the AI for given country.

METHOD:

Country.GetMinRatioToPatchFront (int countryID, int value)

DESCRIPTION:

Checks country's minimum ratio to patch front.

Checks difference between strength value of the units at front.

METHOD:

Country.GetMudBattle (int countryID, int Value)

DESCRIPTION:

Returns 1 if value is correct.

METHOD:

Country.GetMudMove (int countryID, int Value)

DESCRIPTION:

Returns 1 if value is correct.

METHOD:

Country.GetOrientation (int countryID,)

DESCRIPTION:

Checks orientation for given country.

METHOD:

Country.GetPoliticalStrength (int CountryID)

DESCRIPTION:

Returns Political Strength of given country

METHOD:

Country.GetProductionPoints (int CountryID)

DESCRIPTION:

Returns Production Points of given country

METHOD:

Country.GetShipsAmountInSeaZone (countryID, zoneID, shipType)

DESCRIPTION:

Checks how many given country in given seazone possess navy groups.

METHOD:

Country.GetSocialUnrest (int CountryID)

DESCRIPTION:

Returns Social Unrest of given country

METHOD:

Country.GetStrategicMovementPoints (int CountryID)

DESCRIPTION:

Returns SMP of given country.

METHOD:

Country.GetSTP (int countryID, int STP)

DESCRIPTION:

Returns 1 if value is correct.

METHOD:

Country.GetTechInvestment (int countryID, int TechType, int Value)

DESCRIPTION:

Checks tech investment of given tech type type for given country.

METHOD:

Country.GetTechLevel (int countryID, int TechType, int Value)

DESCRIPTION:

Checks tech level of given tech type for given country.

techID parameters:

0: artillery

1: tanks

2: not used anymore...

3: combat aircrafts

4: submarines

5: naval units

6: nuclear weapons

METHOD:

Country.GetTechResearchProgress (int countryID, int TechType, int Value)

DESCRIPTION:

Checks tech research progress of given tech type for given country.

METHOD:

Country.GetUnitCount(countryID, unitType)

DESCRIPTION:

Checks how many units of given type country have.

METHOD:

Country.GetUnitPurchasePreferences (int countryID, int unitType)

DESCRIPTION:

Returns purchase preference value of given country given unit type.

METHOD:

Country.GetUpkeepCost (int countryID)

DESCRIPTION:

Checks upkeep cost for given country.

METHOD:

Country.GetVP (CountryID)

DESCRIPTION:

Checks how many VP country have.

METHOD:

Country.GetWarEconomy (int CountryID)

DESCRIPTION:

Returns War Economy level of given country.

METHOD:

Country.GetWinterBattle (int countryID, int Value)

DESCRIPTION:

Returns 1 if value is correct.

METHOD:

Country.GetWinterMove (int countryID, int Value)

DESCRIPTION:

Returns 1 if value is correct.

METHOD:

Country.GetZoneOfInfluence (int countryID,)

DESCRIPTION:

Checks zone of influence for given country.

******* COUNTRY HAS *******

METHOD:

Country.HasWarWithCountry (int Country1 ID, int Country2 ID)

DESCRIPTION:

Returns 1 if there is a war between two given countries, and 0 if there is peace.

******* COUNTRY IS *******

METHOD:

Country.IsActive (int CountryID)

DESCRIPTION:

Returns if given country is active (countries that surrendered become inactive)

METHOD:

Country.IsAIControlled (int CountryID)

DESCRIPTION:

Returns 1 if given country is controlled by AI, 0 if it is controlled by human.

METHOD:

Country.IsAtWar (int CountryID)

DESCRIPTION:

Returns 1 if given country is at war with any other country. Otherwise 0 is returned.

******* FLAG GET *******

METHOD:

Flag.GetValue (int FlagID)

DESCRIPTION:

Returns the value of given flag. There are 20000 flags available for use.

Values of the flags are of integer type.

For new events added to a scenario use flags with IDs different from events_list file.

******* GAME *******

METHOD:

Game.GetCurrentTurn

DESCRIPTION:

Checks number of turn.

******* HEX *******

METHOD:

Hex.ControlledByAlliance (int X, int Y, int allianceID)

DESCRIPTION:

Returns 1 if hex at coordinates (X,Y) is controlled by given alliance.

Allowable alliance ID values:

1= Allied

2= Axis

3= Comintern

METHOD:

Hex.ControlledByCountry (int X, int Y, int countryID)

DESCRIPTION:

Returns 1 if hex at coordinates (X,Y) is controlled by given country.

METHOD:

Hex.Occupied (int hexX, int hexY)

DESCRIPTION:

Returns 1 if hex with given coordinates is occupied by a unit (land or air), otherwise returns 0.

METHOD:

Hex.OccupiedAndControlledByAlliance (int hexX, int hexY, int allianceID)

DESCRIPTION:

Returns 1 if hex with given coordinates is both controlled by given alliance and occupied by a unit (land or air).

Otherwise 0 is returned.

Allowable alliance ID values:

1= Allied

2= Axis

3= Comintern

METHOD:

Hex.OccupiedByUnitOwnedByCountry (HexX, HexY, CountryID)

DESCRIPTION:

Checks is the hex occupied by a unit of given country. Returns 1 if true

***** SYSTEM GET *****

METHOD:

System.GetRandomNumber (int minRange, int maxRange)

DESCRIPTION:

Returns a random number from given range.

MaxRange should not exceed 2 to the power 16 (65,536)

******* Methods used in <effect> *******

******* AI *******

METHOD:

AI.AddCountryToPress (int countryID, int IdOfCountryToAdd)

DESCRIPTION:

Adds information about country to exert pressure on for AI of given country.

EXAMPLE:

AI.AddCountryToPress (2, 26) means that

AI of Germany (ID=2) should try to exert pressure on Romania (ID=26)

METHOD:

AI.AddDoNotAttackCity (int CountryID, int cityX, int cityY)

DESCRIPTION

Adds city at given coordinates to the list of cities that a given country's AI will not attack.

If there is no city at given coordinates or the city is on the do-not-attack list for AI, nothing happens. List of cities that AI should not attack is defined in ai_data.csv file.

METHOD:

AI.AddExtraCityToGarrison (int countryID, int cityX, int cityY)

DESCRIPTION:

Add city that should be garrisoned by AI of given country.

METHOD:

AI.AddInvasionTarget (int countryID, int cityX, int cityY, int invasion strength)

DESCRIPTION:

Adds new `_land_` invasion target for AI of given country.

Invasion target must be a city, hence parameters are coordinates (cityX,cityY).

City owner should be neutral and the city itself should be a border city.

Last parameter tells AI how much strength to concentrate for invasion.

Setting this to -1 means - concentrate as much strength as possible.

AI of given country can have multiple invasion targets.

EXAMPLE:

AI.AddCountryToPress (2, 134, 110, -1) means that

AI of Germany (ID=2) should concentrate in order to invade

Yugoslavia, because hex (134,110) is the city of Lublana.

Last parameter tells that there is no limit of strength to concentrate on border near Lublana.

METHOD:

AI.AddPortForObligatoryInvasion (int countryID, int portX, int portY, int push Front)

DESCRIPTION:

Adds port city to a list of possible sea invasion targets.

Take a note that AI will make this invasion only if it calculates there is nothing better to invade.

Last parameter tells AI if the port should be added as first on the list (1) or last on the list (0).

METHOD:

AI.AddSeaZoneToPatrolList (int countryID, int SeaZone, int PatrolType)

DESCRIPTION:

Adds sea zone to be patrolled as regular fleet (0) or raider (1).

METHOD:

AI.RemoveDoNotAttackCity (int CountryID, int cityX, int cityY)

DESCRIPTION

Removes city at given coordinates from the list that a given country's AI will not attack.

If there is no city at given coordinates

or the city is not on the do-not-attack list nothing happens.

List of cities that AI should not attack is defined in `ai_data.csv` file.

METHOD:

AI.RemoveInvasionTarget (int countryID, int cityX, int cityY)

DESCRIPTION:

Removes invasion target for AI of given country.

Invasion target must be a city, hence parameters are coordinates (cityX,cityY).

City owner should be neutral, city itself should be a border city and it should be already added to `ai_data`.

METHOD:

AI.RemovePortForObligatoryInvasion (int CountryID, int cityX, int cityY)

DESCRIPTION:

Removes city from a list of possible sea invasion targets.

METHOD:

AI.RemoveSeaZoneToPatrolList (int countryID, int SeaZone, int PatrolType)

DESCRIPTION:

Removes sea zone from patrolling list as regular fleet (0) or raider (1).

METHOD:

AI.SetDesiredStrategicMovementPoints (int CountryID, int desiredSMP)

DESCRIPTION:

Sets desired SMP level for AI of given country.

AI wont purchase any new units until it purchases SMP up to the desired level.

METHOD:

AI.SetDesiredTechLevel (int countryID, int techID, int desiredLevel)

DESCRIPTION:

Sets desired level of given technology for AI of given country.

Only when desired tech level is greater than current level AI will spend PP on research.

This method is very important, using it is the only way to make the AI do any research!

Current levels of technology can be checked in countries.csv, desired levels of techs are in ai_data.csv.

techID parameters:

0: artillery

1: tanks

2: not used anymore...

3: combat aircrafts

4: submarines

5: naval units

6: nuclear weapons

Using techID out of 0 to 6 range is not handled and will result in game crash when firing event.

METHOD:

AI.SetFortLevelToHoldAtAllCost (countryID, newLevel)

DESCRIPTION:

Sets fort level that AI of given country will defend.

METHOD:

AI.SetMakesSeaInvasionsParameter (int CountryID, int Value)

DESCRIPTION:

Sets MakesSeaInvasion parameter for AI of given country.

Value set to 0 means that country will not plan and execute sea invasions.

Any other value means that it will.

METHOD:

AI.SetMaxNumberOfFrontsForSeaInvasion (int countryID, int value)

DESCRIPTION:

Sets number of fronts when AI will plan invasion.

METHOD:

AI.TransportUnitsToFrontsBySea (int CountryID, int value)

DESCRIPTION:

Allows AI country to transport units to fronts or not (0 for no, 1 for yes).

******* AREA *******

METHOD:

Area.ChangeEffectivityOfAllUnits

(int topLeftX, int topLeftY, int bottomRightX, int bottomRightY, int allianceID, int changeValue)

DESCRIPTION:

Changes the effectivity of all units of given alliance in the rectangle area

set by (topLeftX, topLeftY)-(bottomRightX,bottomRightY) to the end of the next turn.

ChangeValue is a percentage change

it may be positive (increased effectivity) or negative (decreased effectivity)

Allowable alliance ID values:

1= Axis

2= Allied

3= Comintern

4= Neutrals

******* CITY *******

METHOD:

City.ChangeProductionPoints (int cityX, cityY, float change)

DESCRIPTION:

Changes Production Points in city at given coordinates by given value.

If no city exists at given (x,y) nothing happens.

Be careful while decreasing Production Points: PP in a city should be ≥ 0 .

METHOD

City.FirePartisans(int cityX, int cityY, int countryID, int unitType, int unitLevel, int unitStrength, name)

DESCRIPTION:

Creates a partisan unit in vicinity of 5 hexes from given city coordinates.

METHOD:

City.SetProductionPoints (int cityX, cityY, float PPValue)

DESCRIPTION:

Sets Production Points in city at given coordinates, if no city exists at given (x,y) nothing happens.

Do not set Production Points to negative value.

METHOD:

City.SetType (int cityX, cityY, int type)

DESCRIPTION:

Changes type of the city.

0 - Main Supply Source

1 - city

2 - MSS with port

3 - port

4 - resources

******* COUNTRY *******

METHOD:

Country.AddConvoy

(int countryID, int convoyType, int value, int departurePortX, int departurePortY, int targetPortX, int targetPortY)

DESCRIPTION:

Adds a convoy of given type for given country.

Set ConvoyType parameter to 0 for supply convoy, and 1 for resource convoy.

If convoy is a resource convoy, use value parameter to set an amount of PP sent otherwise value parameter can have any value.

Set both departure and target port coordinates, if these coordinates are incorrect convoy won't be executed and if they are out of range the game will crash.

EXAMPLE:

Country.AddConvoy (53, 1, 50, 7, 78, 205, 4)

will create a resource convoy for USA, from New York (7,78) to Archangelsk (205,4) carrying 50 PP.

METHOD:

Country.DamageRandomShipsInSeaZone (countryID, seaZoneID, minDamage, maxDamage)

DESCRIPTION:

Damages ships of given country in given sea zone

minDamage and maxDamage must be between <0, 1>

maxDamage >= minDamage

Use dot for decimals

METHOD:

Country.DeclareWarOnCountry (int countryID, int targetCountryID)

DESCRIPTION:

Country with given ID declares war on target country.

All game rules needed to declare such war must be met otherwise event will do nothing.

There is one exception: declaring war by this method does not require any Diplomatic Points.

METHOD:

Country.KillAllAirUnits (int countryID)

DESCRIPTION:

Deletes immediately all air units of given country that are on map.

Use Country.KillAllUnitsInReserve to delete all units in reserve.

METHOD:

Country.KillAllLandUnits (int countryID)

DESCRIPTION:

Deletes immediately all land units of given country.
This includes land units on map and being carried by transports/amphibs.
Use Country.KillAllUnitsInReserve to delete all units in reserve.

METHOD:

Country.KillAllNavalUnits (Country ID)

DESCRIPTION:

Destroys all naval units with land units loaded on transporters and landing crafts

METHOD:

Country.KillAllUnitsInReserve (int countryID)

DESCRIPTION:

Deletes all units (land and air) in reserve for given country.

METHOD:

Country.MakePeace (int countryID, int GiveTerrain)

DESCRIPTION:

Given country finish all wars, and can secede all lost terrains to conquerors (0/1 for no/yes).

METHOD:

Country.NewUnitPurchasePreferences (int countryID, int unitType, intWeight)

DESCRIPTION:

Changes purchase preference of given country for given unit type to given weight value.

METHOD:

Country.RemoveConvoy
(int countryID, int convoyType, int departurePortX, int departurePortY, int targetPortX, int targetPortY)

DESCRIPTION:

Removes a convoy of given type for given country.
Set both departure and target port coordinates, if these coordinates are incorrect convoy won't be deleted and if they are out of range the game will crash.

METHOD:

Country.UnFreezeAllUnits (int countryID)

DESCRIPTION:

Un-freezes all units of given country.

******* COUNTRY CHANGE *******

METHOD:

Country.ChangeAICarrierTargetsPriorities
(int CountryID, int CarrierPriorityID, int CarrierPriorityValue)

DESCRIPTION:

Changes given AI carrier target priority for given priority ID for given country.

METHOD:

Country.ChangeAIP (int countryID, int value)

DESCRIPTION:

Changes AIP value by given amount.

METHOD:

Country.ChangeAllianceEntry (int countryID, int change)

DESCRIPTION:

Changes Alliance Entry of given Country by given value.
If Alliance entry exceeds 100 -
country joins alliance automatically (no further operations needed)

METHOD:

Country.ChangeDiplomaticPoints (int countryID, int change)

DESCRIPTION:

Changes DP of given Country by given value.

METHOD:

Country.ChangeEffectivityOfAllUnits (int countryID, int allianceID, int changeValue)

DESCRIPTION:

Changes the effectivity of all units of given alliance, in the area **ORIGINALLY** controlled by given country (countryID) to the end of the next turn. ChangeValue is a percentage change, it may be positive (increased effectivity) or negative (decreased effectivity)

Allowable alliance ID values:

1= Axis

2= Allied

3= Comintern

4= Neutrals

EXAMPLE:

Country.ChangeEffectivityOfAllUnits (3, 1, -35)

This will decrease the effectivity of all axis units on the territory that was originally owned by USSR.

METHOD:

Country.ChangeMonthsToReachWarfareDoctrineLevel (countryID, int Value)

DESCRIPTION:

Changes how many months needed for new doctrine level.

METHOD:

Country.ChangeOwnershipOfAllNavalUnits (int countryID, int newOwnerCountryID)

DESCRIPTION:

Changes ownership of all fleets & naval units of given country to a new owner.
This includes transports and amphibious!

METHOD:

Country.ChangeProductionPoints (int countryID, int change)

DESCRIPTION:

Changes PP of given Country by given value.

METHOD:

Country.ChangePoliticalStrength (int countryID, int change)

DESCRIPTION:

Changes Political Strength of given Country by given value.

METHOD:

Country.ChangeSMP (Country ID, int Value)

DESCRIPTION:

Changes SMP of given country by given value.

METHOD:

Country.ChangeSocialUnrest (int countryID, int change)

DESCRIPTION:

Changes Social Unrest of given Country by given value.

METHOD:

Country.ChangeSTP (int countryID, int value)

DESCRIPTION:

Changes STP value by given amount.

METHOD:

Country.ChangeTechInvestment (int countryID, int TechType, int Value)

DESCRIPTION:

Changes technology investments for given country, of given type by given value.

techID parameters:

0: artillery

1: tanks

2: not used anymore...

3: combat aircrafts

4: submarines

5: naval units

6: nuclear weapons

METHOD:

Country.ChangeTechLevel (int countryID, int TechType, int Value)

DESCRIPTION:

Changes tech level of given tech type for given country by given value.

METHOD:

Country.ChangeTechResearchProgress (int countryID, int TechType, int Value)

DESCRIPTION:

Changes technology research progress for given country, of given type by given value.

techID parameters:

0: artillery

1: tanks

2: not used anymore...

3: combat aircrafts

4: submarines

5: naval units

6: nuclear weapons

METHOD:

Country.ChangeTerritoryControllerOwnerAndOriginalOwner (idGiver, idReceiver)

DESCRIPTION:

Checks trough the whole map and
if the controller of checked hex is equal to idGiver
then the hex is seceded to idReceiver.

METHOD:

Country.ChangeWarEconomy (int countryID, int change)

DESCRIPTION:

Changes War Economy of given Country by given value.

METHOD:

Country.ChangeUpkeepCost (int countryID, int value)

DESCRIPTION:

Changes country's upkeep cost modifier.
Changes country upkeep cost by given value.

******* COUNTRY SET *******

METHOD:

Country.SetActive (int countryID, int activeFlag)

DESCRIPTION:

Sets if the country is active or not. ActiveFlag==0 means country is inactive, 1 = active.
Countries are by default deactivated when they surrender.
Deactivating manually from event must be done carefully!
Example of deactivating country is Vichy France event.

METHOD:

Country.SetAICarrierTargetsPriorities

(int prior1, int prior2, int prior3, int prior4, int prior5, int prior6, int prior7, int prior8, int prior9)

DESCRIPTION:

Sets all carrier targets priorities at once for all countries.

CarrierTargetsPriorities - Order of priorities:

Attack port garrison when ships are in.

Attack port when ships are in.

Attack port garrison when ships are at sea.

Attack enemy fleet in the same area.

Attack enemy fleet in adjacent area.

Attack enemy air unit in city.

Attack enemy air unit.

Attack enemy land unit in city.

Attack enemy unit in field.

DEFAULT for "CarrierTargetsPriorities" is ;90;80;70;60;50;40;30;20;10

METHOD:

Country.SetAllianceEntry (int countryID, int value)

DESCRIPTION:

Sets Alliance Entry of given Country to a given value.

If Alliance entry is set to 100, country joins alliance automatically (no further operations needed)

METHOD:

Country.SetDesiredLandForcesStructure (int countryID, int unitType, intWeight)

DESCRIPTION:

Sets desired land forces structure for AI for given country at given value.

METHOD:

Country.SetDiplomaticPoints (int countryID, int value)

DESCRIPTION:

Sets DP of given Country to a given value.

METHOD:

Country.SetFreezeLevelForAllFreezedUnits (int countryID, int value)

DESCRIPTION:

Sets Freeze level (number of turns unit should remain frozen)

for units of given country to a given value.

Setting freeze level to 10000 will be displayed to player as "permanent" freeze.

METHOD:

Country.SetMaxCoveringUnitDistance (int countryID, int value)

DESCRIPTION:

Sets maximum covering unit distance for AI for given country at given value.
Sets country's maximum unit covering distance.

METHOD:

Country.SetMaxSafeMoveCost (int countryID, int value)

DESCRIPTION:

Sets maximum safe movement cost for AI for given country at given value.

METHOD:

Country.SetMaxUnitCountOnFront (int countryID, int value)

DESCRIPTION:

Sets maximum number of units in front for AI for given country at given value.
Sets country's maximum units on front counter.

METHOD:

Country.SetMinRatioToPatchFront (int countryID, int value)

DESCRIPTION:

Sets country's minimum ratio to patch front.
Sets minimum ratio to patch fronts for AI for given country at given value.

METHOD:

Country.SetMudBattle (int countryID, int Value)

DESCRIPTION:

Sets mud battle modifier for specific country by given amount.

METHOD:

Country.SetMudMove (int countryID, int Value)

DESCRIPTION:

Sets mud movement modifier for specific country by given amount.

METHOD:

Country.SetOrientation (int countryID, int orient)

DESCRIPTION:

Sets orientation for given country.

Possible choices are:

0 - Firm Democracy

1 - Axis Regime

2 - Communists Regime

3 - Pro Allied

4 - Pro Axis

5 - Pro Communists.

METHOD:

Country.SetPoliticalStrength (int countryID, int value)

DESCRIPTION:

Sets Political Strength of given Country to a given value.

METHOD:

Country.SetProductionPoints (int countryID, int value)

DESCRIPTION:

Sets PP of given Country to a given value.

METHOD:

Country.SetSocialUnrest (int countryID, int value)

DESCRIPTION:

Sets Social Unrest of given Country to a given value.

METHOD:

Country.SetTechInvestment (int countryID, int TechType, int Value)

DESCRIPTION:

Sets tech investment of given type for give country.

METHOD:

Country.SetTechLevel (int countryID, int TechType, int Value)

DESCRIPTION:

Sets tech level of given tech type for given country.

METHOD:

Country.SetTechResearchProgress (int countryID, int TechType, int Value)

DESCRIPTION:

Sets tech research progress of given tech type for given country.

METHOD:

Country.SetUpkeepCost (int countryID, int value)

DESCRIPTION:

Sets country's upkeep cost modifier.

Sets upkeep cost of given country at given value.

METHOD:

Country.SetWarEconomy (int countryID, int value)

DESCRIPTION:

Sets War Economy of given Country to a given value.

METHOD:

Country.SetWinterBattle (int countryID, int Value)

DESCRIPTION:

Sets winter battle modifier for specific country by given amount.

METHOD:

Country.SetWinterMove (int countryID, int Value)

DESCRIPTION:

Sets winter movement modifier for specific country by given amount.

METHOD:

Country.SetZoneOfInfluence (int countryID, int zone)

DESCRIPTION:

Sets zone of influence for given country.

Possible choices are:

0 - none

1 - Allies

2 - Axis

3 - Commitern.

******* FLAG *******

METHOD:

Flag.ChangeValue (int flagID, int change)

DESCRIPTION:

Changes value of given flag by given value.

METHOD:

Flag.SetValue (int flagID, int value)

DESCRIPTION:

Sets value of given flag.

******* HEX *******

METHOD:

Hex.AddAirUnit

(int countryID, int hexX, int hexY, int unitType, int unitLevel, int unitStrength, string name)

DESCRIPTION:

Adds an air unit for given country to pointed hex or to unit pool (if hexX and hexY =-1) for given country with given type, level, strength and name.

METHOD:

Hex.AddLandUnit

(int hexX, int hexY, int countryID, int unitType, int unitLevel, int unitStrength, string name)

DESCRIPTION:

Adds a land unit to pointed hex or to unit pool (if hexX and hexY =-1) for given country with given type, level, strength and name.

METHOD:

Hex.AddLandUnitAndPassControlOverHexIfEmpty

(int hexX, int hexY, int countryID, int unitType, int unitLevel, int unitStrength, name)

DESCRIPTION:

Adds a land unit to pointed hex for given country with given type, level, strength and name.

If at hex already other unit is presented, nothing happens

If hex is controlled by other country then a control over hex goes to countryID

METHOD:

Hex.AddMulberry (int countryID, int hexX, int hexY, string name)

DESCRIPTION:

Adds a mulberry unit for given country to pointed hex, with a specific name

METHOD:

Hex.AddNavalUnit (int countryID, string name, int unitType)

DESCRIPTION:

Adds a naval unit for given country with given name and type.

METHOD:

Hex.ChangeFortLevel (int x, int y, int Level)

DESCRIPTION:

Changes fortification level of given hex.

METHOD:

Hex.KillUnit (int hexX, int hexY)

DESCRIPTION

Kills the unit that occupies hex with given coordinates.

If no unit is present on the hex - nothing happens.

If the coordinates are out of range, game will crash, so be careful.

METHOD:

Hex.SetFortLevel (int x, int y, int newLevel)

DESCRIPTION:

Sets fortification level of given hex.

Be careful, range is not checked.

Entering incorrect hex coordinates will cause game to crash.

Fort level should be in range from 0 (no fortification) to 9 (max fortification).

METHOD:

HexStripe.ChangeControllerAndOwner (int hexX, int hexY, int stripeHeight, int newOwnerId)

DESCRIPTION:

This method changes controller and owner of a vertical hex stripe, beginning from hex with coordinates (hexX, hexY).

Hex stripe height is specified by third parameter (stripeHeight), hexes are changed from start hex going south.

Owner and Controller of hex is changed to a country specified with 4th parameter (newOwnerId).

Additional effect: each unit not owned by specified country or its allies and standing on changed hex is automatically transferred to its owner's reserve pool.

EXAMPLE:

```
<effect method="HexStripe.ChangeControllerAndOwner" param1="10" param2="24" param3="3" param4="7" />
```

Three hexes: (10,24) (10,25) and (10,26) will change their owner and controller to country with id=7.

METHOD:

HexStripe.ChangeControllerOwnerAndOriginalOwner
(int hexX, int hexY, int stripeHeight, int newOwnerId)

DESCRIPTION:

Identical to HexStripe.ChangeControllerAndOwner but this one also changes original owner of hex stripe.

******* PORT *********METHOD:**

Port.ChangeSize (portID, changeValue)

DESCRIPTION:

Changes size of the given port by given value

METHOD:

Port.SetSize (portID, newValue)

DESCRIPTION:

Sets size of the given port to the given value

******* SEA NODE *******

METHOD:

SeaNode.AddLandUnit

(int nodeID, int navalUnitType, int countryID, int unitType, int unitLevel, int unitStrength, string name)

DESCRIPTION:

Add a naval unit with a land unit loaded.

Note that naval unit must be a transport or amphibious assault.

METHOD:

SeaNode.AddLandingLandUnit

(int nodeID, int navalUnitType, int countryID, int unitType, int unitLevel, int unitStrength, string name, int portX, int portY)

DESCRIPTION:

Deploys naval unit with loaded infantry unit ready to assault at given hex.

METHOD:

SeaNode.AddLandUnitWithDest

(int nodeID, int navalUnitType, int countryID, int unitType, int unitLevel, int unitStrength, int destNodeID, int aiOrder, string name)

DESCRIPTION:

Creates a troop transporter at given sea node with given land unit and with destination set.

METHOD:

SeaNode.AddNavalUnit (idSeaZone, idOwner, name, type, currentHP)

DESCRIPTION:

Creates naval unit in given sea zone. And optionally current HP of unit.

******* UNIT *******

METHOD:

Unit.Freeze (int unitX, int unitY, intTurns)

DESCRIPTION

Freezes unit at given coordinates for given number of turns.

If no unit is present at given coordinates nothing happens.

METHOD:

Unit.Remove (countryID, type, unitID)

DESCRIPTION:

Removes a unit from map. For random choice use -1.

METHOD:

Unit.Teleport (int sourceX, int sourceY, int destinationX, int destinationY)

DESCRIPTION:

This method teleports unit from hex (sourceX,sourceY) to (destinationX,destinationY).

If no unit is present at source hex or destination hex is occupied - nothing happens.

NOTE: teleportation does not check who controls destination hex

and after unit is teleported destination hex controller remains unchanged!

METHOD:

Unit.Unfreeze (int unitX, int unitY)

DESCRIPTION

Un-freezes unit at given coordinates.

If no unit is present at given coordinates

or present unit is not frozen - nothing happens.

******* VIEW *******

METHOD:

View.CenterAt (int hexX, int hexY)

DESCRIPTION:

Centers map on given hex
